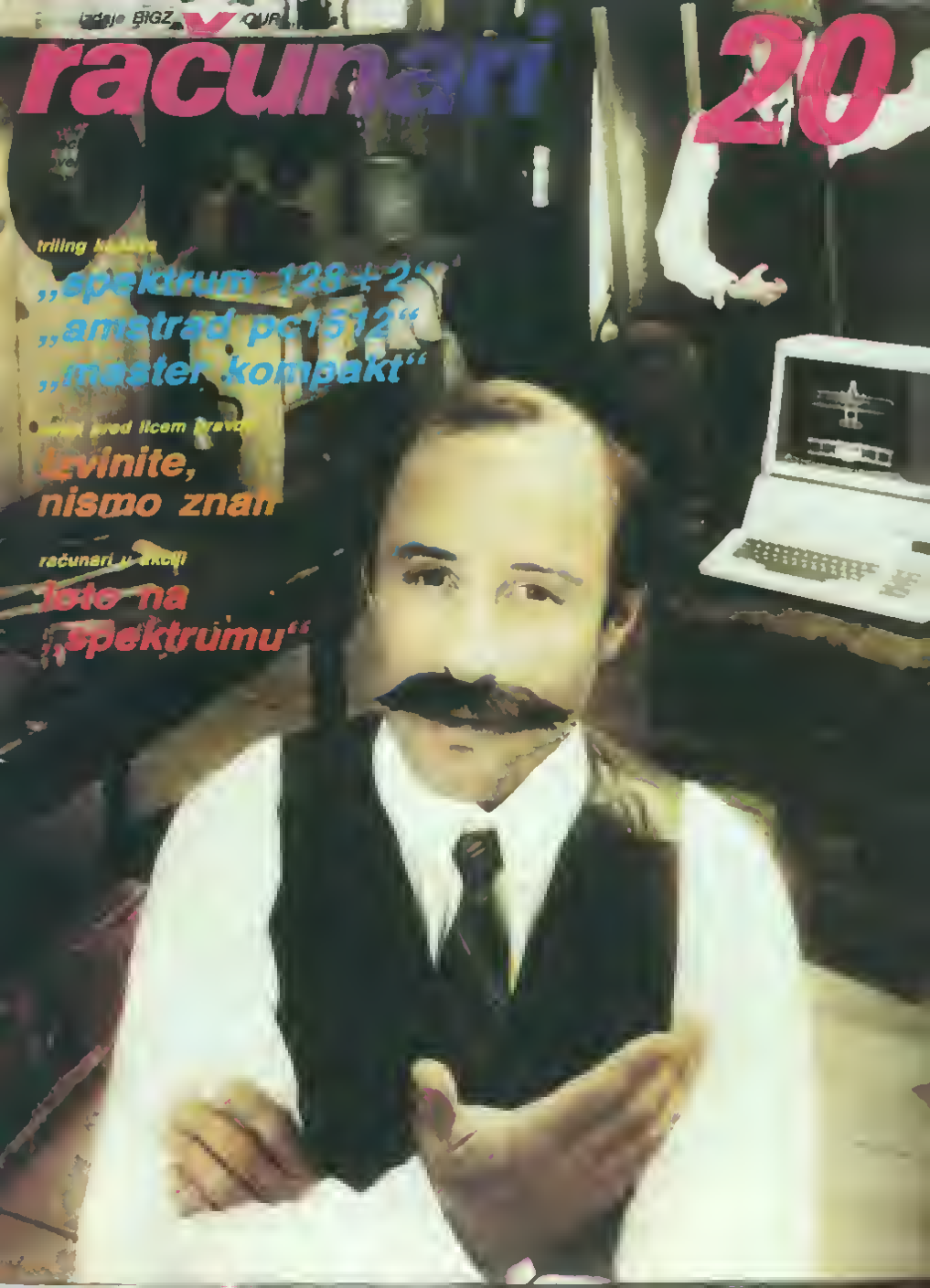
izdaje BIGZ ... OUR...

računari

20



triling kasete

„spektrum 128+2“
„amstrad pc1512“
„master kompakt“

... pred licem pravde

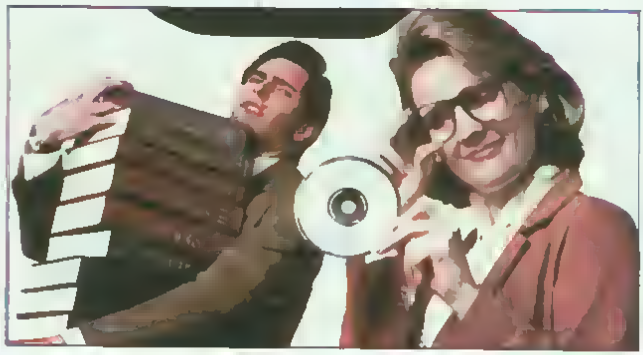
Izvinite, nismo znali

računari u akciji

loto na „spektrumu“

Dr Radomir
A. Mihajlović

Razglednica iz Njujorka



Optičke memorije

Filips i Du Pon su nedavno o... marketing i tehnički centar ze Sjedinj... Država se ciljem da zajedničkim snegam... nastupe na emeričkom tržištu optičkih me... morija

Laboratorija za spacijelne tahnička epli... kacija optičkih memorija se kancelarijama za marketing smeštana ja u Wilmingtonu država Dalavar u novoizgrađenim prostori... ama od 1200 kvadratnih metare U istom ...adu se nelazi i prva febrika optičkih ROM ...skova U stilu svih velikih avropskih kom... panija koje zbog raspoloživosti reprometa... ...jala i tahničkog personala otvaraju labri... ...ke visoke tahnologije na taritoriji SAD ...ilips je uspostavio identičnu proizvodnu ...ju u Ajdholanu Holendije

Svoju verziju optičkih disk jedinica Filips ...nazvao CD-PROM Za razliku od stan... dardnih CD-ROM diskova u koje sa infor... macije upisuja laserskim graviranjem udu... ...lenja ne disku CD-PROM jadinice upisuju farmaciona bitova promenom faze povr... ...skog metarijala diske Filipsov uredaj ja ...stanju da upisuja i očitava podetke kako ...sa CD-ROM tako i sa CD PROM diskova

Proizvodnja CD-ROM diskova ja skuplja ...solati se samo u velikim količinama dok proizvodnje CD-PROM diskova biti ako... mski opravdana i u menjim količinema Očekuja se da Filipsove CD-PROM jedi... a košta menje od 1 000 dolere e sam ...sk manja od 100 dolera

Glavni kupac Filipsovih optičkih diskova ...a sade Digital pozneti proizvodeč mini... ...a

Laserske zdravstene legitimacije

...ja Kalifor... ...poznatom i mom Kanon ...enu se masovnom proi... ...zdravstvanih legitimaci... ...rojekat vrednosti multimi... ...alo ga je nekoliko deča ... dvojice jugoslovanskog ...Sarić)

...deja gološradih momaka ...da se kartica sa laserski ...ma ogromnog kapaciteta ...pis kompletna zdravstvene ...kartice donela ja svakom ...pomoć advokata i komšije ...ko četvri miliona dolare U ...ronmentu kade je kriza sa kontrolom infor... ...acija o lošковima lačanja nekoliko milio... ...a osiguranika poalela avidentne glavna kompanija za zdrevstvano osiguranja u SAD Plavi Krst i Plevi Štit (Blua Cross end Blue Shield) prihvetila ja se odušavljanjam idaju dačaka o upotrabi laserskih legitima... ...cije

Primana novog načine identifikacija i evidantiranja pecijanata omogućiće osigu... ...ravejućam društvu iz države Meriland da proširi broj osigurenika se per miliona na 80 miliona Osnovni uslov Plavog Krste i Plavog Šlita pre kompletnog prihvatanje ove nove tahnologija je bio de Draksal izabera još jednu kompeniju koja bi kao rezervne proizvodile legitimacija u uradaja ze očitavenja U Americi ja nalma prekse de sa svaki projakat velikih dimanzije obe zbadi sa više proizvodeče istovatne opre... ma Ovakvom poslovnom politikom sa izbe... ...geveju krizna nastaćice reprometerijele a i spračave se mogućnost ucana Rezervni proizvodeč ja medu mžanjarima poznet popularno keo drugi izvor (second source)

Odluka Kraksela da drugi izvor bude Kanon bazirana ja ne činjanici de je Kanon jedini bio u stanju de u potpunosti garantu... ...je totalnu raspoloživoal bilo kog broja legitimacija u bilo kom momantu zadovo... ...ljavejući kvalitet i spramnost da se brzo prilegodi novim zahtevime za modifikaci... ...jom legitimacija ako ovi kojim slučajam skrsnu u budućnosti

U upotrabi je tranulno 15 000 aksperi... ...mentalnih legitimacija Laserki čiteći su ...izvodnja Kanon po licenci Drerkalara ...avl Krst i Plavi Štli tekođe poseduje licencu Glavn... ...osoci svih ...ih prava ...akak...

Teraatari

Meda je još uvak gotovo neverovetno zamisliti lični računer se memorijom od jednog gigabajta (miljerde bajte) uskoro se mogu očakivati memorije hiljadu puta većeg kapecilata rade veličine tarabajta TB Holandska firma DOCdata planire leser... ...ski sistem se optičkom trakom kapacitata jednog tarabajte Prolotipni uradaj DOKto... ...čak (DOCwhaal) koristi 128 optičkih kaseta veličine popularnih audio kasala i kapacita... ...la 8 gigebajta (8GB) erenžiranih u vidu kruge površina 0 3 kvadratne matre

Varuje se da će DOKločak biti u stanju da pronađe bilo koju detotaku izmedu astronomski velikog broja mogućih datoteka ze samo 10 sekundi

Komarcijalna raspoloživoat terabajtnih mamorija je planirana ze 1987 godinu Mogućnost da sa samo jednom upisuja ne optičku traku DOKtočka ne predstavija zne... ...čajan handikap u odnosu na izbrisivu mag... ...netnu traku Sekvencijalni karaktar upisiva... ...nja ne treku ne dozvoljava česta izmane upisenih datotaka kao što je to slučaj se magnetnim diskom (raorganizacija datote... ...ka ne treci uzima dosta računaskog vrame... ...ne leko da sa i magnetna traka koristi gotovo kao netzbrisivi madijum) Neće nas narevno iznenediti eko jedan od prvih računera bude poneo Ime taragalaksija ...ti baram teraatari

Mikroprocesor od galijum arsenida

...donel Daglas uspeo ja da integriše 1860 ...galijum-arsenidnih tranzistora na četvoro... ...struki četvorobitni mikroprocesorski čip MD 2901 MD-2901 je prvi mikroprocesor napravijan u novoj tahnologiji

Optička memorija ogromnih kepacitata i komunikacioni kenali sa optičkim vlaknima ogromna informaciona propusne moći za... ...hteveju velika brzine rada pratećih elak... ...tronskih (u budućnosti varovatno komplat... ...no optičkih) uredaje Zalo se u Sjedinjenim Državeme intanzivno trega za alternetivnim rašenjime Rezultat višegodišnjeg istraživa... ...nja prvenstvano u SAD predstavljaju gali... ...jum arsanidni elektronski sklopovi sa velike brzina rada i male potrošnja elaktrična energija Primeta redi prvanec MD-2901 ...sa svaga 135mW Radi se o procesoru ...a bit-kniška (bit slice), koji uspašno ...sava programe nepisane ze poznati ... 2901 mikroprocesor

...laboratorijama Mekdonal Deglasa ja u ...22 hilnog galijum-arsanidnog

Softverske pljeskavice

Kompenija Borland internecionel poste... ...la je preko noći jedan od glavnih proizvo... ...đača softvara za mikroračunare Na Konfe... ...renciji za automatska i kompjutarska maši... ...na (ACM) u Ostinu Takasa 1985 godine

autor je imao prilike da se sretne za ovnivačima ove danas značajna kompanija Idaja grupe relativno mladih momaka iz grada Skots Veli, Kalifornija bila je da za piretima dosekoči niskim cenama softvara toliko niskim da za više isplati kupiti softvereki paket za garancijom nego ga kopirati Broj pirata u odnosu ne ragularne kupca softvara pokazao za zapanjujući Naima broj prodatih prvih Borland-ovih Turbopaskal paketa po ceni od 39 doiara bio je 10 prema 1 u odnosu na broj prodatih paskal prevodioca bilo koja druge firma, čija su cena do izlaska Borlanda na tržišta bila minimum 100 dolare Borland Internacional je bila prva tirma koja je sa parolom „što jeftiniji softvar-to više kupaca u atilu slavnog Makdonalda (proizvođač jeftinih lapinja I pljeskavica) postala softvarski gigant

Dosledni avojoj poslovnoj politici ocenivači Borlanda au nedavno otkupili malu softversku firmu Singular-Sotver sa avim proizvodima i projaktima, da bi avim proizvodima Singulara odmah oborili cenu za 33% Singular je osnovan pre godinu dana Jedini uspešni proizvod je bio paket „Interlaja za relacionu obradu baze podataka za Eplov „makintoš" Za nepunih godinu dana Singular sa svega sedam zapolanih zaradio je prodajom ovog pakata preko milion dolara Borland je „Interlaju prilagodio I za IBM-PC I prodaja ga pod nazivom „Reflaks" Cena „Reflaksa" je 99,95 dolara

Prva elektronska biblioteka

Američki Institut Elektro I Elaktronakih inžanjara (IEEE) je juna ova godina instalirao u Nju Jorku prvu alaktronaku biblioteku Na konferanciji za štampu potpredsednik zaktore za publikacija IEEE Čarls Haus demonstrirao ja pratraživanja 40 000 atranica avih publikacija objavljanih 1984 godina uz pomoč običnog personalnog računara Damonstracija alaktronakih publikacija predatavlja prvi korak ka budućem totalno alaktronizovanom distribuiranju tehničkih informacija Izgleda da ja papirnim publikacijama Gutembergovog tipa potpuno odzvonilo

Količina tehničkih informacija uvećave za avakodnavno neočekivanom brzinom Kako zabreti ave rzapoložive podatke o nakom naučno-tehničkom problamu u akonomski opravdanom vremanskom periodu I kako raziučiti aktuelne od pravazidanih činjanica, Na ovo pitanje ja jedino moguće odgovoriti kompjuterizovenim pratraživenjam I obradom Na drugoj atrani au anormni troškovi klasičnog štampanja avega što naučno-tahnički autori žela da asopšte zaintaresovanim čitaocima Kurioziteta radi za nedaljno izdanja Nju Jork Tajmza polrabno ja poseći gotovo neverovetnih dva atotina hektara šuma?! Mnogobrojna publikacija IEEE, distribuirane po celom avetu zahtevaju najverovatnija daleko više papira Da se kojim alučajem ne recirkuliše alari papir, mogli bismo samo da samisilmo kako bi Amarika I šumovita Kanada izgledali za pedesetak godina

Fabruara ova godina na sastanku generalnog rukovodatva IEEE odlučeno ja da za započne sa ‚soft-publikovanjam' Predložana au dva odvojana ali srodna projekta Izgrednje alaktronake baze avih do asda podataka objavljanih I izgrednja aistema za

podnošenja novih članaka za publikovanja alaktronakim putem alaktronakom poštom ili alanjam diakata za tekstom Za sada zamo kompjutareka zakcija IEEEa prima predložene takstove u softvaru Nedostatak dorečenog standarda u vezi za formatom članaka uz format računarskog komunicirania kao i nedovoljan kvelitat postojeće kompjuterske grafike za prihvetljivom cenom koštanja osnovna su prepraka alaktronskog zabirenja predloženih rukopiza

Prenos rukopisa za jednog na drugi računar raziličitog tipa još uvek nije jednostavan zadatak Pra tri godina ja pri udruženju amerických Publicata (Azsociation of American Pubiishere AAP) započeto sa razradom standardnih procedura za alaktronako podnošenja članaka za štampu Rezultat dosadašnjih napora je odštampan ove godine kad Standardni generalni jazik za označavanja (Standard Generalized Markup languaga SGML) Korišćenjam SGMLa autora sada mogu da lično koduju svoja rukopiza Prednost korišćenja SGMLa ja ubrzano objavljivanja članaka bez napotrabnih prakucavanja i komplikovanog slaganja nepripremljanog tekata

Na sastanku u avguatu ove godine garnerairo rukovodatvo IEEEa je razmatrelo datalja alaktronskog tzv telakontarizanja Razmatrane tema au bile alaktronako publikovanja bibliotaka budućnoati kompjuterizovano konfarisanja grafičko uakladivanja I buduće matode kompjuterizovenog tehničkog obrazovanja

Jedan od konkratnih projakata koji su u toku je kreiranja indakaa predavanja za avih (nakoliko atotina) naučno-tehničkih konferancija održanih 1985 godine u SAD Početna investicija je 100 000 dolare Do januara ove godine prema rečima Hauza, avi pretplatnici na računareku mrežu „Kompjutereka pošta (Compma I I), ili na mrežu Nalaženja svoga puta (Find-Your-Way), biče u mogućnosti da korišćenjem parsonalnog računare i modama pretražuju indaka za 1986 godinu

Da bi za lapunila baza podataka od gore pomenutih 40 000 stranica tekata upotrabljen ja optički akaner za direktno učitavanja podataka koji au zatim, upizani avi na zamo jedan laserski diak od 12 inče Demonstrireni alatam ja razvijan u firmi Univerzity Mikrofilm Internacional (UMI) iz En Arbore država Mičigan koja poseduja kopija avih mgiataraklh i doktorskih disertacija ikada napisanih u Sjedinjanim Državama. Predvidja za da će UMIev aistem biti povezan sa korisnicima fakaimil mašinama visoka gratičke rezolucija Naima predvida sa mogućnost kompjuterskog pratraživanja tahnička baze podataka i štampanja samo žaljanih tekstova i grafika

Po zaduženoati gotovo avih zamalja sveta kod amarićkih benaka da se dalimično zaključiti od kolikog je značaja savršenost i afikasnost manipulisanja akonomakim informacijama Informacija o promanama situacija na svetskom tržištu su dostupne amarićkim biznismanima znatno ranija ka šnjanjem nego bilo gda u svetu Očigledno na redu je automatizovana obrada na zamo akonomskih već i naučno-tahničkih informaclja Po mišljanju mnogih, zamlja industrijalizovanog zapada na čelu sa SAD informacionom akaplozijom aksponancijalnom brzinom uvačavaju tahnološki procep u odnosu na srednja i zamlje u razvoju Ostavimo budućnoati da odgovori kakve će implikacije ovo imati na čovečanstvo.

Klon sa japanske strane

Svojom verzijom IBM-PC AT računara, za oznakom MBC-990, Sanlo za nedavno pridružio armiji proizvođača AT kompetibilnih računara Za razliku od ostalih klonova. Sanlon MBC-990 ja računar minimalne konfiguracije bez ikakvih ugrađanih opcionih ploča Kupcu za ostavlja alaboda da ugraduja prama potrebi ploče za različitim funkcijama. Standardna konfiguracija za 512K RAMa flopi disk jedinicom kapaciteta 1,2 MB I operativnim siatemom MS-DOS 3,10 košta 2 599 dolare što ja više od cena po kojima akvivalentne računara od 30MB kolor monitorom Sanlo-CRT-60 i kolor karticom cena MBC-990 se panja do 4 000 dolare

3/razglednica iz njujorka

Spektrum 128 plus 2

„spektrumovo" novo ruho

Na ovogodišnjem PCW sajmu u Londonu, koji je održan početkom septembra, „AMSTRAD" je predstavio najnovija izdanja starog, dobrog „spektruma", koje je krstio „+2", i pustio ga u beli svet . . .

Novinari skoro svih angisakih računarskih časopisa su gotovo poludeli od sreće! Kompjuterska revije su prepuna novih slika prikaza, iznutrica, hvalcapeva, a sve je to posvećeno „Sink pardon, „Amstradovom novom pulanu, „+2

O. K. idemo redom Prvo .

TASTATURA

Može sa reći da je oblik i raspored tastere upravo najbolje kombinacija starog „plusa" i „CPC 464 . Svi tasteri su prilično zgodno raspoređeni, jedino je „adit" previše blizu „caps shiftu" pa ponekaoa možemo ulateti u editovanje i bez svoje volje. Takođe taster „break je u gornjem desnom uglu gde je prirodno mesto za „dalate" tako da ponekad prati sami polete prema čošku Ipak, oni koji su navikli na raspored kod „plusa", ovde će sa naći na svom tlu

Najvažnije karakteristika nove tastatura je da nema štampanih tastere se „keywordima", što se dosada najviše i zamaralo svim Sinklerovim tastaturama U stvari, ostala su samo tri, i to „run „code" i „load" — valjda zbog toga što se najviše upotrebljavaju Ipak, treba imati na umu da će sa novi „spektrum" koristiti najviše u modu 128, a u mod 48 ćemo sa vrećeti samo radi igara ili ponekog specifičnog programa. Uostalom, mnogo je lakše kucati slovo po slovo, nego juriti šiftove naoklo Zato sam najviše voleo „Beta Basic"

Još jedna stvar — razmaknica je postala nešto veća, jer su i svi tasteri prošireni Lepo, pogotovo za ona kojima „spektrum" služi za pisanje

Ah, da, osećaji imam gadan utisak da se ispod krije membrana, ali ovoga puta sofisticirana sa osećajem kao kod „amstreda CPC 464 ali sa blagim klikom Perfektno idemo dalje

KADA SE UKLJUČI . . .

„ izgleda gotovo potpuno isto kao i običan „spektrum 128K" Ipak, pada u oči natpis „ © 1986, © 1982 Amstrad Consumer Electronic plc" što gotovo nagoni suze na oči, a i nagovešteva velika problema sa određenim programima koji testiraju čeksum ROM-a, jer je sada produžena reklamna poruka Takođe, u meniju nedostaje opcija „tape tester", jer ugrađeni kasetofon (verovatno) na može da omana

Sve ostalo uktjučujući i atribute i ostale skalamerije ostalo je po starom, tj blesavo i drago (ljubi ga majka)

KASETAŠ . .

predstavlja prvu stvar koja upada u oči čim se vidi novajlija Da li će učiteti sve programe, na znamo ali znamo da nema podešavanje jačina ili boje tona ako stvari krenu naopako, jedino možete da uzmete šrafciger i da pokušate da podesite glavu kroz mali otvor sa gornje strana Dalje, prva i jedina spolja uočljive MANA je ta da kasetofon NEMA ponavlje NEMA BROJAČ!!! Nazdravlje! Zamislite situaciju da sve ona brižljivo sortirana i obalažena kasete više na mogu da sa lako upotrebljavaju, jer gde li ono beše „Tasword na prvoj ili drugoj trećini strana? Zaista, „Amstredovci" su mogli da se malo sažala i odvoje para za lola bolju verziju kasetofona koja bi imala brojač, pa makar to koštalo preko 150 funti Inače Englezi tvrde da je ovaj model bolji od onoga ugrađenog u CPC 464 ! Robusniji je i ima debiju pogonaku treku Ako malar tera, onda na pomaže ni automobilska guma

UNUTAR KUĆIŠTA . . .

leži čudna zverka. Za onoga koji nikada nije otvorio stariji „128 , nove štampana pločica će izgledati u prvi mah kao IBM-ov „motherboard" dobro, dobro, praterao sam Međutim, ima novosti Hladnjak je sada opet iznutra, samo bolje smešten, a i famozni kalorifer „ULA" je dobio hladnjak na leda. Jedina razlika se vidi na čipovima U sred srede leži novi čip koji pompezno nosi oznaku „Amstred — to je u stvari, novi ROM (sa novom kopirajt porukom!) — a na msatu gde stoji oznaka za verziju pločica piše — dobro, sve već znate

ULAZI/IZLAZI . . .

na sve strane Prosto je neverovatno koliko otvora je dobio novi „spektrum" Ako sa setimo 1982 godina i starog „gumenjeka" Ovako, i dalje je tu konektor opšte namene, ali ima problema sa njim (čitaj dalje) Tu su i RS 232/MIDI interfejs (vrlo slični konektori kao kod QL, u stvari britanski PTT dškovi), utičnica sa „key-

ped", RGB monitor, TV prijemnik, izlaz za pojačalo (juhuuuuli!), tu su i reset taster i dva porta za džojstik.

E, pa baš se tim džekovima nešto nije u redu ! Ne kutiji računara iznad ulaza lepo piše „koristite samo naše ! SJSI džojstike". Ovo već miriše na „Amstrad" i njegove „dva u jednom" džojstike, ali ne drugi način. Naravno, džekovi su naizgled isti „ateri" standard, ali tek će pažljivo oko uočiti razliku u razmaku između kontakata. Originalni „SJS1" džojstik liči na „Quickshot", ali košta oko 15 funti i ni po čemu ne zaslužuje tu čast da bude kupljen. Podvala koju je zamislio Alen Šuger-Šećerko nije bila dugog veka.

Čim su se vratili se PCW sajma, inženjeri firma „Cheetah" su napravili adapter se „normalne" džojstike i ugradili ga u svoj dobar džojstik 126, koji je tom prilikom dobio „+" u imenu i sada se prodaje kao „126+" i košta samo 9 funti! Pravda ipak pobeđuje.

Da, i još jedne stvari. Džojstici rade po protokolu „Interface 2", tako da moramo reći zbogom starom, dobrom, univerzalnom „Kempstonu".

PROGRAMIRANJE. . .

je potpuno isto kao kod „128K" modela. Jasno je da svojim izgledom tastatura tera na programiranje u modu 128, ali i za one koji ne žele tu mogućnost, uz računar je priložena tabela „ključeva", po kojoj možete da za setite gde „ ono beše STRS, i ASSI?"

KOMPATIBILNOST

Evo, otprilike, šta se (tvrdi) oko tog fakta

— svi priključni uređaji i periferalije bi trebalo da rade bez problema. Jedino, zbog debljine plastike kod konektora opšte namene, mikrodrajvi bi trebalo da se pažljivo uguraju šrafcigerom iii sličnim alatom isti problem će postojati i kod ostalih uređaja.

— Softver bi kampletno trebalo da bude kompatibilan, tj ako je radio ne „128K", onda ni , +2" ne bi trebalo da ga odbija. Jedini problem je kod programa koji testiraju čeksum ROM-a ; već sam rekao

— Stari džojstici neće pasovati ne novu mašinu, dok se ne pojave (pojavili su se) novi uređaji prilagođeni novom obliku konektora

U najgorem slučaju (zar?), možete utaknuti svoj stari interfejs i koristiti ga kao i kod prethodnika. Sa konje ne magarca, ali šta se tu može !

U SVAKOM SLUČAJU. . .

Za 150 funti dobijate zaista mnogo . . Mašinu koje sigurno ima budućnost, jer iza nje ne stoje više masočari koji svoje prve ideje daju na tržište, već oštrijni poslovnjeci, koji svaki peni triput prevrnu pre nego što ga ulože u nešto Računar nasleduje i ogromnu biblioteku od skoro (III preko) 5000 programa, što predstavlja veliku prednost.

Jedna od najvažnijih činjenica je ta da nijedan proizvodač računara ne nudi ovakav sistem za ovakve pare, ni „Ateri", ni „Amstrad" a pogotovo ne „Komodor". Držimo palčeve se starog kraija, jer njegovo novo ruho to zaslužuje!!!

Darko Stanojević

5/spektrumovo novo ruho

Šta ima novo

Šetanje na daljinu

Sigurno ste čuli za Panmanov ploter To je onaj ploter koji se umesto velike ploče i komplikanog uređaja sastoji samo od jedne kutijice i dugačkog kabla Kutijica se sama kreće po stolu (i po papiru) i cetvrtja za sobom normalan trag Zgodno? Sad je još zgodnija. Pojavila se varijanta Panmanovog plotera u koji je ugradan modam Šetajući ploter se lepo priključi preko telefona ne vaš kompjuter koji je kilometrima daleko, i crta šetajući po stolu Za sad ne znamo cenu (B Đ)

Amstrad vedri i oblači

Neverovatno ali istinito! Samo od marta ove godine „Amstrad" je uspeo da proda oko 250 000 komada računara 8256 i 8512i Mnoge konkuranteka firma su pokušala da prate ovaj uspeh obaranjem cena svojih računara ali do danas niko nije uspeo da prestigne, iii samo dostigne Amstradov bum

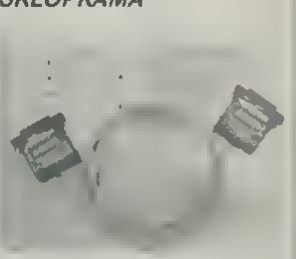
Dalje, firma je odlučila da svoje goraopomenute računare poekupi (!!i) za 5 funti, napačivanjem priručnike za Meilardov bejzik, koji se do sada besplatno dobijao uz mašinu Razlog za ovu dodatnu dopistu je, prema rečima službenika iz Amstrada, vrlo jednostaven samo jedan od deset kupaca je koristio bejzik za programiranja mašine, pa je odlučeno da se priručnik prodaja odvojeno III po većoj ceni

Varovatno zbog toga, bejzik programi se još uvek naiaza na B atrani sistemskog dieka kao CP/M fajlovi (D S)

Matrični ploter

imate PCja Želite da crtate i kostrišeta na njemu Zatljubite se u mogućnceti nekog CAD programe Nabavite ga od nekog prijatelja (naravno- besplatno) Šta vas tada preseče? Varovatno činjenica da bi vam za dostojno prikazivenje rezultate vašeg crtanja trebao ploter koji nije samo odvrtano skup, već ga je jednostavno nemoguće kupiti jer ata poslednja para dali na matrični štampač. Jedini vaš izlaz predstavlja EOgraph pioča EOgraph ja pioča koja ne samo da služi kao jako koristan printarski bafar (iako bi samo za to bila malo preskupa) nego omogućuje da vaš matrični printer izbacuje crteže za koje ja veoma teško reči da ih nija nacrtao ploter. Cena ploča je negde ispod 600 funti (B Đ)

RS 232 KABEL SA SKLOPKAMA



Pri povezivenju računala sa vanjskim jedinicama može doći do raznih problema, od kojih su neki i nestandardni priključci Da bi se takve smetnje izbegle tvrtka Lindy iz SR Njemačka proizvela je RS 232 kabel sa ugrađenim sklopkama, tako da kod eventualnih preinaka nije potrebno odlemijivanje i ponovno lemljenje, već je dovoljno podesiti željenu komunikaciju samo uz pomoć sklopki Dužina kabela je 2 m Cijena je relativno niska — 70 DM (Z V)

Ispomoć za „amstrad"

Program Sidekick za PC Kompatibilne računara već je postao prava legenda uprkos (III možda upravo sehvaljujući) svojoj jednostavnosti I oblčnosti Seda se na tržištu pojavio i njegov prepev za „amstradove računare pod imenom „Write Hand Man" Program košta 30 funti i može da bude sasta koristan Pored svih „Sidekick'-ovih performansi, ima i mogućnost da „amstradovu tastaturu privramano tokenizujete

Rukavica u lice

Nekim kompanijama nikako ne polazi za rukom da postanu besteseri Međutim, to ih ne sprečava da i dalja usavrševeju svoje modela Najnoviji primari su firma „Tatung" i „Saga Systems", koje se trude da pariraju „Amstradu

„Tatung je proizveo novi računar nazvan „Tatung 256 koji kao što mu i samo ime govori, ima 256 K memorije 14 inčni color monitor i odvojenu tastaturu Dalja, ima 512 boja, i stereo zvuk. Bazican je na Z80 procesoru i naravno, CP/M kompatibilan Firma ga ceni ravno 400 funti

„Compilment je novi, prvi računar firme „Saga Systems" i po rečima predstavnika te kuće, ima da bija Amstrada na rodenom terenu! To je Z80 sistem (opet!) koji će koštati oko 300 funti znači 100 funti jeftinija od PCW 8256. Sistem će obuhvatati 3 5 inčni diek-drajv matrični štampač brzina 100 karaktara u sekundi i gomiiu poklon-softvera Monitor nije uključen u cenu, ali, ako se i on ubeci u računicu, cena opat nije viša od „Amstradove"

„Saga Systema" je poznatija po dodatnim tastaturama za „spektrum i bilo bi lepo kada bi ove nova mašina ličita ne naku od njih (O S)

Šta ima novo

Čavrljajte sa Tajvancem

Ako slučajno imate tajvance (više nismo toliki optimisti da pretpostavljamo da neko ima čak i IBMovog PCja) možete ga naučiti da sluša vaše komande zamerke psovke i bilo šta što biste želeli da mu kažete To vam omogućava Vocalink-paket koji se sastoji od pločice mikrofona i diskete Ova kombinacija omogućava vašem PC-kompatibilcu da razume oko 500 različitih komandi od kojih već dobijate neke pripremljena za korišćenje Wordstara i Lotusa Prava stvar za lenjivce Cena? Prava stvar Negde oko 425 funti (B Đ)

„Oric" sa ponovo!

Firma WE Software ja odlučila da produži samrtničke muke računara ORIC (nemamo ne kažemo „Oric-Nova pogodite zašto!) time što je ponudila njegovim vlasnicima nekoliko periferijskih uređaja Da vidimo šta se tu kuva

Prvo disk-jedinice sa poboljšanim Sedoric operativnim sistemom dvostruke gustine pakovanja po ceni od svega 239 65 funti što predstavlja novost jer je do sada njegova cena bila za 30 funti viša Ah da format diskete je 3 inča šmrc- žalosno

Drugo, modem To je model V23 koji pristiže zajedno sa interfejsom i programom za podršku treće- printer „kosmos i četvrto programabilni interfejs za džojetik.

Mnogo poznatog (?) softvera će pristići sa redukovanim cenama što će jako obradovati sadašnje vlasnike tog vremenom pregaženog (ne)popularnog računara (D S)

Nova cena „Računara"

Proizvodni troškovi „Računara", naročito zbog poskupljenja papira, poslednjih meseci su toliko narasli da redakcija više ne može da ih podnosi sama. Od ovog broja, kao što je i red, dali mo ih se svojim čitaocima Dakle od sada za trećinu akupije i, nadamo se, barem isto toliko bolje

REDAKCIJA

DG/One 2

Tvrtka Data General, izbacila je na tržišta po karakteristikama i dizajnu vrhunsko prijenosno računalo Pun naziv novog računala je DG/One Modeli 2 Ono se može povezati na velika Data General uredska sistema, kao i na IBM mainframe i PC kompjutore

U osnovnoj varijanti računalo ima 256 K memorije koje se može proširiti do 640 K Ugrađena je i 3,5 inčos disk jedinica, a operacijski sustav je MS DOS 2 1

U računalo su ugrađeni serijski i paralelni interfejsi, a može se dodati i tvrdi disk kapaciteta 10 megabajta ili flopi disk jedinica za 5,25 inčne diskete U samom računalu po atoji još dovoljno mjesta za aventualna proširenja, npr za procesor 8087

Ekran ja poboljšane LCD verzije Cijena osnovne konfiguracije računala je cca 5300 DM, a sa tvrdim diskom i procesorom 8087 cca 9600 DM Ukoliko se želi kvalitatniji EL-ekran (elektro lumiscentni) traba dopiatiti još 2600 DM

Upravljenja rijačima

U SAD ja predstavljan novi sistem za upravljanje računala IBM PC AT rijećima Korisnik računala preko mikrofona i posebnog interfejsa unosi riječi (tekst) u računalo Cijeli postupak je znatno kvalitetniji od prvog ozbiljnog sistema te vrste koji se pojavio 1984 godina Slične naprave su postojale i ranije ali kvaliteta nije zadovoljavala visoka zahtjeve za profesionalnu upotrebu

Zanimljivo je da je čip koji omogućuje spomenuto upravljanje računala rijećima, razvijen u Švicarskoj To je tzv DSP-Chip (Digital Signal Processor)koji se sastoji od dva podsistema kojima upravlja AT procesor DSP čip može izvesti oko 50 miliona operacija u sekundi

Sistem rada je takav da korisnik prvo , unese rijećima probni tekst, računalo PC snimi govornu karakteristiku te uskladi prijam u računalo Tak potom se prelazi na unos glavnog teksta

Prvi podsistem pretvara govorom unešeni tekst u znakove (fonetske simbole) a drugi podsistem od tih znakova tvori cijela riječi kojih u sistemu ima oko 5000 Za to je potrebno oko 2 megabajta memorije Pouzdanost rada je vrlo velika — iznad 95 posto (Z V)

Feniks za modele

Prava stvar za inžanjera — program „Phoenix 3D za projektovanje trodimenzionalnih modela koji se zatim mogu rotirati emanjivati povećavati, rastezati seći i deformisati na sve moguće načine Jako zgodno zar ne I sve to za samo 39,95 dolara ako se obratite na adresu Dreams of the Phoenix Inc - PO Box 10273 Jacksonville Florida, USA (B Đak)

Stamparija na stolu

Dobro ne baš štamparija ali bar slovoslagačnica. Dok se mi nadamo sličnim stvarima u Engipakoj vlada ogromno interesovanje za malo nezavisno izdavaštvo uz pomoć kompjutera. Zato se od ekora mogu naći paketi opreme potrebne za tako nešto po relativno niskim cenama Tako „mekintoš plus', dodatni drajv od 800K LaserWriter laserski štampač, MacDraw, MacPrint, Majkrosottov tekst procesor Word program za uređivanje novine Aldus PageMaker i par konktora i pomoćnog materijala mogu da se nađu za nato malo prako 6500 funti To je možda ogromna suma za naše krizne krajeve ali je sasvim razumna ako se ima u vidu posao koji ona znači Razmislite, prodajte plac i kola i postanite nezavisni izdavač (B Đak)

Bez virenja

Sećate li se naše vesti iz „Računara 15 o tome kako je neki holandski naučnik usevršio matod pomoću koga se iz blizine vaše zgrade pomoću opreme koja košta samo 15 dolara može otkriti šta piše na ekranu vašeg monitora ili TVa Ako vas je to nateralo da se zabrinete, možete odahnuti Firma Securitec je razvile specijalnu vrstu pokrivača za ko pjuter i ekran koji sprečava zračenje pomoću koga je holandski naučnik Vim Ven Ek postigao iščitavanje Taj pokrov ili prekrivač će najesen početi da proizvodi kanadska firma Erintek a trabalo bi de košta oko 500 (USA) dolara (B Đak)

20

računari

izlazi jedanput mesečno
cena 400 din. novembar 1986.
Specijalno izdanje
časopisa „Galaksija"

Izdaje
Beogradski izdavačko-grafički zavod
OOUR Novinska delatnost Duga
11000 Beograd
Bulevar vojvode Mišića 17

Telefoni
650-161 (redakcija)
650-528 (prodaja)
651-703 (propaganda)

Generalni direktor
Dobrosav Petrović

Direktor OOUR Duga
Bratoljub Babić
Glavni i odgovorni urednik
Gavrilo Vučković
Urednik izdanja
Jova Regasek

Tehnički urednik
Mirko Popov

Redakcija časopisa „Galaksija"
Tanasije Gavranović pomoćnik glavnog i odgovornog urednika
Esad Jakupović, zamenik glavnog odgovornog urednika
Aleksandar Milinković, urednik
Jova Regasek, urednik
Zorka Simević, sekretar redakcije
Srđan Stojančev novinar
Gavrilo Vučković glavni i odgovorni urednik

Stručna saradnja
Dejan Ristanović
Dušan Slavić
Nevenka Spalević
Anđelko Zgorelec

Spoljna redakcija
Branko Đaković Dejan Ristanović Jelena Rupnik Jovan Skuljan prof dr Dušan Slavić Nevenka Spalević Zoran Životić

Stalni saradnici
Nada Aleksić Ninoslav Čabrić, Branko Đaković Voja Gašić, Branko Hebrang Đorđe Janković, Vladimir Kostić Vladimir Krstonošić Radomir A Mihajlović Zvonimir Makovec Blažimir Miše Dejan Muhamedagić, Ivan Nador Radomir Nikolajev Zoran Obradović, Miodrag Potkonjak, Dejan Ristanović, Jelena Rupnik, Dušan Slavić, Jovan Skuljan Nevenka Spalević Darko Stanojević, Zvonimir Vlatrička, Anđelko Zgorelec, Zoran Životić

Izdavački savet „Galaksije"
Dr Rudi Dobjanši prof dr Branislav Dimitrijević (predsednik) Radovan Drašković, Tanasije Gavranović Živorad Gligić Esad Jakupović Velizar Maslać Nikola Pajić Željko Perunović prof dr Momčilo Ristić Vlada Ristić dr Inž Milorad Teofilović Vidojko Veličković Velimir Vesović, Milivoje Vuković

Štampa
Beogradski izdavačko grafički zavod
11000 Beograd, Bulevar vojvode Mišića 17
Žiro-račun kod SDK 60802-833--2483
Devizni račun kod Beobanke 60811-620-8-82701-999-01068
Za inostranstvo cena dvostruka (400 D 2 50 US$ 6 50 DM 45 Sch 5 50 Sfra 20 Ffra)
Na osnovu mišljenja Republičkog sekretarijata za kulturu broj 413--77/72-03 i „Službenog glasnika broj 26/72, ovo izdanje oslobođeno je poreza na promet.

sadržaj

2/ razglednice iz njujorka

3/ računari u izlogu
„spektrumovo" novo ruho

4/ šta ima novo

8/ ioad „dragi računari

9/ računari u izlogu
„amatrad pc 1512"

12/ računari u izlogu
„master kompakt"

15/ peek & poka show

16/ komercijalni softver
turbo je nešto drugo

18/ komercijalni softver
treća areća

19/ računari iz mog ugla
Kako sa oženiti i živeti srećno atotinu godina

20/ mikroprocesori
procesor naših anova

23/ dejanove pitalice

24/ računari i pravo
izvinite, nismo znali

25/ hakerski vodič kroz beograd
u potrazi za izgubljenim vremenom

27/ umetak
6502

43/ mali ogiasi

47/ tehnika programiranja
crtanje na mašincu (2)

50/ tehnika programiranja
kula bez rekurzija

52/ programiranje na mašincu
sinklerova veza

55/ kontroverze
algoritam „brzi gonzalesa"

56/ tehnika programiranja
iz šupljeg u prazno

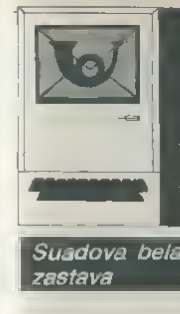
60/ u domaćoj radinosti
turbodrajv za „spektrum"

62/ računari u akciji
loto na „spektrumu"

64/ u domaćoj radinosti
korak ka „plusu"

65/ šta ima novo u svetu komponenata

66/ razbarušeni sprajtovi



Suadova bela zastava

Dragi moji vozdra!

Pošto sam bio pošteno naružen od strane vaših čitalaca (spekirumovaca uglavnom) i posebno od vašeg saradnika Željka Jurića i zato što sam malo razmišljao čitate ovo pismo. Dakle, zakljućio sam da je rat između spekirumovaca i bejtovićevski nastrojenih komodorovaca ili kao i rat između Iraka i Irana. Ni jedni ni drugi nikad neće priznati da je njihov računar lošiji od onog drugog ali će i jedni i drugi vazda tvrdili da je njihov računar bolji od onog drugog a to kao što vidimo nema nikakvog smisla i zato spešljam (zar Suad Bejtović začetnik javnog rata?) na ave spekirumovce a posebno ne komodoriste da zakopaju ratna sjekira i operu se od ratničkih bojа. Evo prvi ja se izvinjavam (da li dovoljno javno?) svim spekirumovcima za svaku uvredu na njihov i ne račun njihovog računara. Posledice ne objavljivanje ovog pisma mogu biti

1. Šta to opet iskombinuje Jaj Bejtović? Moj računar je bolji i tečka!

2. Vidi šta je to ovaj Bejtović dobro smislio! Nije baš moj računer najnaji! Nije ni onaj toliko loši

O kvalitetu časopisa više se ne ma šta pričalii. Sprajtovi još naj bolji ali mi se najviše sviđa članak

Pletonove ljubavne muke. Osiao sam bez teksta. Strašno! Majo me zbunjuje to što je Božidar Markuši ć ozbiljno shvatio hakersku priču pa posilno scenario po kojem se stvarno može napraviti igra. Ali nama veze nije ni to loše napraviv

E imam i ja nešto i za Usijani joystick na radosti onih koji misle da šamaram džojstik 10 sati ne dan i da imam biblioteku od barem 600 programa. U jednoj od novijih igara za komodor Uridijumu posti gao sam skor od 69285 poena i doputovao do 7 nivoa koji se zove PLATINA. Pošto niko nije siao niže ja toj igri smatram da je to rezultat za divljenje. Ili možda neko drugi ne smatra?! Ja sa ne ljutim kad je neko bolji od mene. Zalim. ušteđi sam strastveni ali mi se u galaksi jama baš ništa nije dogodilo. Od Harmless do Compatenia. Uh da bio sam dikatno pripušten da li ću uzati najredu stvar u univer zumu ili sam čitaoci Računara odgovorio se NOOOO i izbjegao problema sa zvrkovima. Da čujemo samo. Na lijepom plavom Dunavu ! Savršenstvo! Znam da iako ne misle vlasnici spektruma 128 ali se mora priznati da komodor (svaka čast 128 BASIC-u ima izvjadbi da čovek siuča i da se divi. Jel' tako? Jedno pitanje! Kako (ti ne koju adresu) mogu da se prijatim ne PCW? Hvala. Evo kao što se vioi ostalo mi je još kap-dve tinte u naliv-peru te vas lijepo pozdravljemi (mrzi me da milenjam patronu) i želim što manje ratničkih pisama

Vaš,
Suad Bajtović
Sarajevo

Mnogo priznat da svakve pukajničke pisma od redovnih čitalaca ovih luča ekа računara nismo očekivali. Naravno, to ne znači da Suadov postupak nije ispravan čak supretno! On je lepo otpočeo proces, video šta je napravilo, posuo se pepelom izvinuo... I šta još hoćete! Očekujemo dalje po jedno svakvo pismo od kapitalne ekipe spektrum i , amstrad , a ni , BBC dim ne bi bio lanj, kada bi se malo potrudile da se spuad , iz oblaka

Sdi kojim Suad plaže je vrlo lep i interesantan i zato bi bilo bolje da se avrši pametnijim atverima, nego šta su prepucavanja. Zato, avi vi ljaterl pišite nam ali svaga puta sa konkretnim atvarima kritikama opisima igara, svojim i prerađenim programima vaših i si. Vrata su vam uvek otvorena

O K na kraju ave adresa PCW-a

STANOJEVIĆ DARKO

Mirjanin okrugli sto

Drage računerkinje zdravo!

Javljam vam se već ireći put sa nedom da ćete okončati rat koji vodite sa vašim čitaocima. Popelo mi se na vrh glave to što u svakom lanju (a to je poslednjih pet-šest) čliem

— C-64 je super gluposali ništa ne valja spekirum je gluposi živeo komodor i slično

Meio lužno deluje da dozvoljavate deci da se na ovaj način prepucavaju prako novina izgleda mi da je njihov jedini cilj ući ne stranice na brzjući sradstva Iskra no rečeno ni vaša redakcije nije napravila dobar potez što nastevaje io prepucavanja i dalje. Jednostavno treba okončati s limi

Što se drugih tema iiče nema ni iu mnogo pohvale ipak dopada mi se Biblioteka knjiga. Mislim da ste napokon našli mesto gde ćete informisati ljude o knjigama. Ali ne bi trebalo da se zadržite samo na toma. Treba upuiiti neći kritike na neka izdanja kao što su raznorazni priručnici za programere lipa. Kako opaleli pravi programer. To mi zaista deluje kao tabu lisme. A i kada spomenete to je sve lugaljivo i svodi se samo ne inicjate (A K) Jeanije glasnije drugovi!

Postoji jedne lužne stvar o kojoj novine nerado pišu. Svi komentari šu knjiga za osnovnu i srednju školu valjda zato što su one naj masovnije a niko ne pominje studente — visokoškolce. Smešno je kad kažem da sam udžbenik, u oktobru 1985 piaiila i000 a da će nova cena biti preko 4500 dinara. Da li vi imate komentara na ovo? Tužno

imam još jedan komentar. Mislim da nije u redu da se danes prelazi prako činjenice da razno razni radnički univerziteli prave programere i operatere za zamisiite, 3 6 ili 9 meseci. Kao ne fabrič koji iracl. Pa za koji davo ljudi uče u srednjoj školi programiranje programeka jezika (bajsik kobol forlran ave ili pojedinačno), programaske sisteme i onoiika matematika?!

Uvek je neš nerod volieo pričicom pa makar arijed u gluposi. Radnim organizacijama se to normalno više isplati jar ne može da otera dugogodišnjeg radnika koji posteje maltene tehnološki višak nego ga lepo obući za male pare i zdravo. Vuk sli i ovce ne broju. Pa nije ierl A to isto važi i za mene koja završavem višu školu (za primenjenu informatiku i statistiku) i imam 14 predmeta. Smrači mi se pred očima

No iako te visokoumne ustanove (fakulteti, instituti) i organizuju nekakav kurs-seminar iu pravo prilatupa imaju samo izabrani ili zaposleni. Ali moliću lepo zašto se i iu na bi uzele para — uplati kolizaciju ako si studani (eko možeš?!)

Moj predlog je da se organizuju besplatna predavanja poiedinih profesora ili siručnjeka koji su ra spoloženi da pranose znanja na mlada narašiaje. Naravno da bi ovaj potez uspeo mora imati pokrovitelja. Predlažem da to budete vi. Šta kažete na predlog?! Ma genijalan je

Ako vam je potreban dežurni kritičar ovog vašeg-našeg liste samo se javite na donju adresu

Puno pozdrava od

Mirai
Mirjane Nikolić
Bulevar revolucije 255/5
i i050 Beograd

Obredevali sma za kada sma nakon tolika pauze, primili tvoja pisma. Odmah sma ga poznali — ti znaš po čemu. Primetia te naravna za dežurnog kritičara. U našoj redakciji žena ienska uglavnam kritikuju a muškarci uglavnom atvaraju list. Navrad će redakcije da se upoznema dtvarma

Uspešna zamena

Od pra nekoliko brojeva sam primatio da imate sve manje teksto va druga Jovane Skuljane šio smatram za propust jer su njegovi čtenci jedni od najboljih koji se objavljuju u vašem časopisu. Zašto ja to teko?

ipak smatram da nisia baš sva uprskali. Primetio sam da od skoro kod vas radi i drug Željko Jurić čiji su tekstovi u vazi s spekirumom iakode vrlo interesantni. O čemu se radi? Da li se Jovan umorio od pisanja i da li ste Željka ubacili kao zamenu?

Srdačan pozdrav
Pavlović Slaven
Sisak

Jeca i dalje ne posredan način, služi Računerima i našim čitaocima. Nalme- en je na odsluženju vajnog roka i čim završi sbuku ova će biti po starem Željke je navi čtan u našoj ekipi. Bez obzira na obzira

Vratite ženske

Pročitzo sam najnovije Računera. Nema potrebe da nabrajam neka tu pohvale jer vi verovatno znate koliko ste dobri. Zato pravo na temu naslovna! Da se razumemo — ja sam jako želeo da znam kako izgleda amstradov PC i ne samo to one slika sa onim ljudima u mafijaškim odalima je jako dobro ispale ipak pored svega toga vratite one vaše ženske na naslovnu stranu. Možda ženska nisu uvek najlepše ali su vam zato naslovne uvek najzanimljivije. Još jednom — vratite ženske!

Rodoljub Indić
Sevska 47
Beograd

Muški deo redakcije (mi ame u večini) prihvatio je tvo predlog sa oprezom. Šta kažeš za ovu?

Nova tarifa

Napisaću vam p.smo u samo sedam rečenica. Ono što meni smate u vašim pismima su predugačka pisma koje pošalju pola rubrike i onda ostane malo prostora za zaista zanimljive stvari. Ta pisma sigurno pišu skribomani koji se odluče kad vide svoje ime u novi nama. Nek se oni dopisuju među sobno a vi im slobodno sečite pisma u male malecke delove ili ih bacajte u koš. Predlažem da svi koji vam napišu pismo duže od četiriseći reči moraju da za reči koje su višak plate tarifu (kao za mala oglasa). Tako ćete dobijati kraća pisma a možda ćete nešto i zaradili. Puno pozdrava Vukosiav Bajkić
Sarajevo

Jeste Vukoslave, bile je same sedam rečenica ali sedam duuugih rečenica. Blažeme se sa tvojim predlogom. U tvom pismu je bile tačne 58 reči (ima ti štampame besplatno). Kad odbijeme 40 reči, ostaje 18 teke da nam ti duguješ isman za jednu dobru večeru. Javi se

Sopstvena rubrika pisama

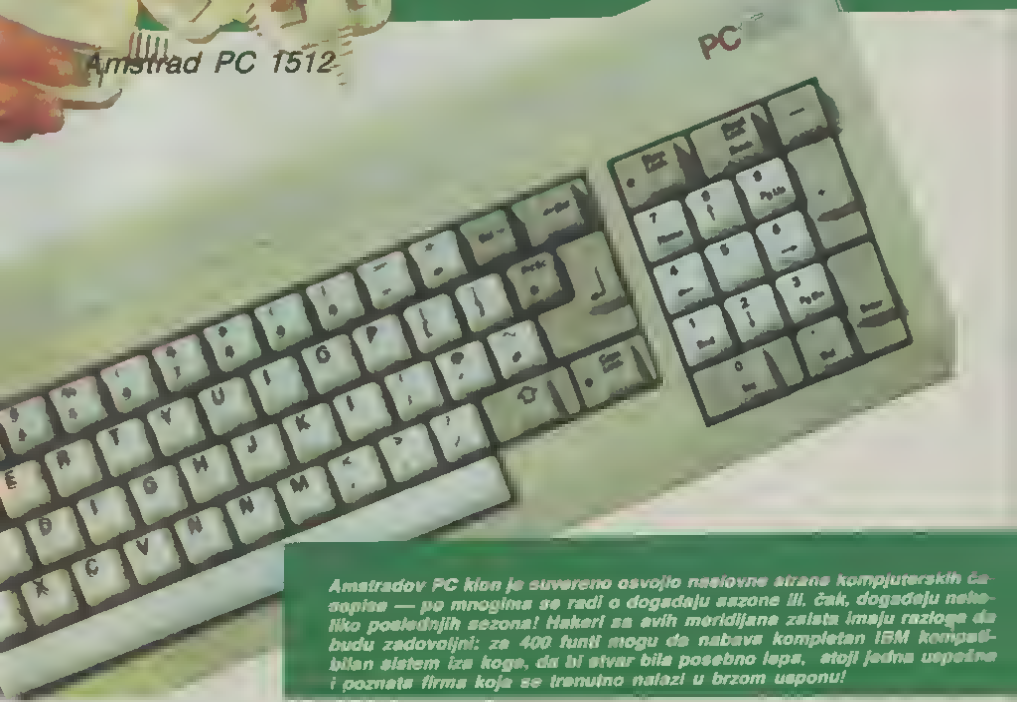
U prosiet (u Raču a a ste objavili jedno minjaturno si smo koje se ilcelo Peek & poka show-a! Tek kada sam njega video u rubrici Load Dragi Računeri shvatio sam da veoma ratko objavljujete pisma koja se tiču P P Show-a. Ako je sudti po interesovanju koje moje društvo pokazuje za tu rubriku mora de vam stuže puno pisama povodom nje. Da li ih vi namerno ne objavljujete ili vam te pisma nisu odvoljno interesantna? Ako su mekr približno u sliiu Peek & poka show-a zasiužuju da budu objavljena. Uzgred mogu li ja da pošaljem koji prilog za P PS? U svakom slučaju primite puno pozdrava od

Jevsra Stoiljkovića
Novi Sad

Dragi Jovane, pisma koja stižu povodom PEEK & POKE SHOW a mogu na podeliti (grubo) u dve grupe veća grupa ne nastoji iz pisma ianašenih obeležaiaca, dok se manje ili konkretnije grupe zastoji od mrtvinam nabijenih zapisa — Onih Drugih . I jedne i druge bi ezbiljne narušile ukočeni akademski nivo koji pokušavame da dame našem cenienom i skupom listu, pa ih ne objavljujeme

šećerni pc

Amstrad PC 1512



Amstradov PC klon je suvereno osvojio naslovne strane kompjuterskih časopisa — po mnogima se radi o događaju sezone ili, čak, događaju nekoliko poslednjih sezona! Hakeri sa svih meridijana zaista imaju razloge da budu zadovoljni: za 400 funti mogu da nabave kompletan IBM kompatibilan sistem iza koga, da bi stvar bila posebno lepa, stoji jedna uspešna i poznata firma koja se trenutno nalazi u brzom usponu!

Iako verujemo da su osnovne karakteristike Amstradovog PC-ja već dobro poznate u hakerskim krugovima poput svih IBM komatibilaca amstrad PC se sastoji od tri dela tastatura osnovna kutija u koju je ugradena štampana ploča i disk jedinica i koja predstavlja podnožje za treći deo monohrom (ili kolor) monitora Dimenzije su prilično mala računar će zauzeti samo 0 15 kvadratnih metara vašeg stola!

Tastatura je mehanička i, kada se uzma u obzir cena izuzetno dobra sastoji se od 59 QWERTY tastera deset funkcijskih dirki i numerička tastatura od 16 dirki Nlau obezbeđeni posebni tasteri za pokretanje kurzora ali se odgovarajući efekat postiže pomoću numerička tastatura što se vidi i sa slike Tastatura je sve u svemu radena po uzoru na IBM PC AT ali je napravljen jedan neobjašnjiv klik veoma lepi i veliki tasteri ENTER SHIFT ISERT i slični su niži od ostalih i dopunjani površinom za pritiskanja veličine oblčnog tastera Uzvišenja na tasteru vam nikako neće pomoći pri brzom kucanju iako je najasno zašto ‚amstrad PC nema normalna kontrolne tastatura tsetetura novog PC klona može da zaseluži solidnu ocenu

Amstrad se pobrinuo da tastutura ne bude jedini način za upravljanje kompjuterom uz računar ćete dobiti i argonomski oblikovanog miša se dva tastara koji je podržan postojećim operativnim sistemom Miš se međutim, neće pokazati naročito konsno upotrabijavaće ga pri radu se GEM-om ili nakim programima koji budu pisani specijalno za ‚amstrad PC . ali će većina čuvenih programa za IBM PC i dalje primati komanda samo sa tastature

Mikroprocesor(i): sasvim klasičan izbor

Mosek računara predstavlja Intelov mikroprocesor 8086 koji je semanio ‚originalni 8088 koji koristi IBM Ove promena trabe da nas obraduja 8086 je softveraki 100% (zaista 100%) kompatibilan sa 8088 ali ima šesnaestobitnu magistralu za podatka što znači da omogućave brži rad Amstrad PC' uz to, radi na kloku od 8 MHz što znači da bi trabao da bude 2 puta brži od IBM PC-ja (4 77 MHz)

Da li ugradnja bržeg mikroprocesora može da dovede do problema sa kompatibilnošću? Ako biste pokušali da prodajete spektruma koji rade sa Z80 B na ktoku od 6 MHz, ne biste napravili neročit posao lonako prateške igra bi postale toliko brze da niko ne bi ni pomišljao da ih igra pa bi novi model uz svu kompatibilnost, za trenutak ostao bez upotrabljivog softvara' IBM PC se druga strana, ne predstavlja mašinu koja se kupuja zbog igara — poslovni softver će brže raditi ako posedujete brži procesor ali će obe verzije računare biti sasvim upotrebljive Zamanu mikroprocesora je olakšao i sam IBM koji u svoje novija modela (XT i AT) ugradija 80186 i 80286, a razmišlja i o računarima sa 80386 — najveći deo postojećih programa će dakle perfaktno raditi bez obzira na brzinu mikroprocesora

Zašto Amstrad kada je već manjeo procesor nija odabrao 80186 čip koji koriste IBM? Razloge traba tražiti u cenama 80186 je i dalje relativno nov čip koji uz to nija beš lako nabavlti u velikim količinama na tržištima Dalekog Istoka gde sa Amstrad snebdeve (kako onda IBM rizikuja? Veliki Plavi je toliko važan kljent da će Intel uvek najpra podmiriti njegove potrabe Amstrad ja najzad verovatno pnmetio da

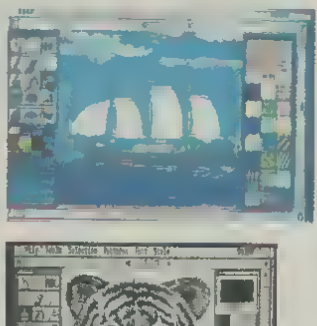
sa 8086 sasvim dobro pokazao kod Olivetijevog M24 i mnogih drugih kompatibilaca pa ga je izabrao kao savršen kompromis između mogućnosti i cena

PC 1512 ima 512 kilobajta RAM-a koji se na ploči može proširiti do 640 kilobajta prostim umetanjem memorijskih čipova. Znajući da je 640 kilobajta najveća memorija do koje se standardni IBM PC proširuje većina softverskih firmi projektuje svoje proizvode tako da se izvršavaju baš u ovoj konfiguraciji. Preporučujemo vam, dakle da proširite RAM do krajnjih granica jsate će će mnogi komercijalni programi raditi i na mašini sa 512 kilobajta ali ćete pra ili posle naići na paket koji vam životno treba i koji na vašem računaru ne radi. Ako živite u Londonu, proširićete kroz nekoliko radnji i dokupiti memoriju. Ako međutim živite u Beogradu, ova će vas hitne kupovine malo skupje koštati!

Kada smo već kod praznih podnožja Amstrad je omogućio i ugradnju aritmetičkog koprocesora 8087. Šta je uopšte aritmetički koprocesor i da li ga treba kupiti? Kod bazmalo svih kućnih računara glavni mikroprocesor, uz pomoć softvera u ROM-u sabira oduzima množi i deli racionalne brojeve i izračunava vrednosti elementarnih funkcija. Zar to nije sasvim normalno? Jeste ali se u poslednje vrama radi i nešto bolje razvijeni su specijalni mikroprocesori koji služe **samo** za obavljanja aritmetičkih operacija sa racionalnim brojevima i računanje vrednosti funkcija. Obzirom da je softver koji upravlja ovim računanjam pisao u mikrokodu, dakako je brži od svakog mogućeg mašinskog programa. Osim toga aritmatički koprocesor u nekim situacijama radi paralelno sa centralnim 8086 čima se dalje ubrzava rad. Ispitajte investiciju? Samo ako strašno mnogo računate u mnogim primenama vrama potrebno za računanje predstavlja pravu aitnicu u poradanju sa vremanom naophodnim za komunikaciju sa diskom ili neka druga slične posiove. Obzirom da je 8087 relativno skup čip (oko 120 funti) nećete pogrešiti ako se za početak odreknete njegovih usluga.

Grafika, jedna čudna odluka

Kada govorimo o IBM klonovima pitanja grafike je veoma značajno. IBM je naime, u početku prodavao računara sa takozvanom MDA *(Monochrom Display Adaptor)* karticom koja je omogućavala rad sa tekstom na crno-belom ekranu. gratika jednostavno nije bilo jer se smatralo da ona nije potrabna za posiovna primana računara. Ubrzo se pokazalo da sa parsonalni kompjuteri itakako ozbiljno koriste za posiovnu gratiku CAD/CAM i mnoga druge stvari koja zahtavaju crtanja po ekranu a često i crtanja u doji (i dalje je doduše prisutno mišljenja da je za većinu posiovnih primane dovoljna crno-bela gratika jer se slika u boji teško prenose na papir i docnija teško umnožava ju) pa je IBM pripramio CGA karticu — *Colour Graphics Adaptor*. Za IBM-ovu kolor karticu se obično na naiazi baš mnogo kompiimanata, amatra sa da je razolucija po vertikali (200 tačaka) slaba da su slova ružno dizajnirana i da sa sporo iscrtavaju da je izbor boja pramali — ipak većina posiovnog softvera koji je napisan za IBM PC podrazumeva postojanja jedne od dve pomanute kartica što znači da sa nija lako odlučiti za kupovinu neka dailimično kompatibilne video ploče superiornih karakteri-



Gem Paint

stika. Mnogo dalajnju argumentaciju za razna video kartica možete da nadate u tekstu *Prava slika za prave programere* iz računara 19.

Šta je Amstrad mogao da uradi? Žalio je računar koji će saradivati i sa kolor monitorom, pa je morao da odustane od *Hercules* kartice i IBM-ovog MDA. Nove IBM-ove kartice kao što su EGA *(Enhanced Graphics Adaptor)* i PGA *(Profesional Graphics Adaptor)* su verovatno bila preskupa da bi ih imalo smisla kopirati, pa je amstrad PC snabdeven video interfejsom koji je kompatibilan sa IBM-ovom CGA karticom. To praktično znači da je moguć rad sa gratikom 320*200 u 4 boja ili 640*200 u dve boja, boja su tiksna i ne biraju se iz palete. Što se teksta tiče u svakom od 25 redova sa po žeiji korisnika može lapisivati 40 ili 80 znakova. Amstrad PC može da se koristi i sa monohrom monitorom pri čemu će boja biti zamanjena nijansama sivog.

Pošlo smo završili sa tehničkim karakteristikama grafike pogledajmo kako one mogu da utiču na kupovinu Amstradovog PC kompatibilca. U poslednje se vrama amatra da je za kvalitetno crtanja potrabno imati bar 512 tačaka po vertikali ekrana što znači da su kose linija na amstrad PC-ju tseterasatija nago što je dopustivo. Traba, ipak da primatimo da IBM PC (pa samim tim i njegovi klonovi) nisu baš idealni za izuzetno praciznu grafiku i brzu animaciju — ako planirate ovakva primana razmislite o „amigi". Za standardnu posiovnu gratiku i „obične" CAD primana gratika 640×200 predstavlja sasvim prihvatljivo rešenja. Na kraju krajava za pranošenja gratika veća razolucija na papir traba posadovati veoma dobar (i veoma skupi ploteri).

Navolja se CGA karticom su slova većina ljudi koristi IBM PC za rad se tekstom i bazama podataka što znači da kvalitet slova i brzina skrolovanja ekrana predstavijaju ključna kriterijuma za izbor grafička kartice. IBM je prilično namarno dizajnirao slova pa su obiici nekih od njih previše „stepenasti". Slova, da bi stvar bila posebno nazgodne bljeskaju dok ekran skroluje što je vrlo neprijatno pri dužam radu.

Amatrad je rašio veći od ova dva problama: ekran je stabilan i relativno lepo i brzo skroluje. Slova su druge strana nisu naročito lapa ali sa sa njima može (i mora) živeti. Možda ste pomislili da opremite PC 1512 *Hercules* karticom i koristita ja za rad sa tekstom? Na žalost, neće ići vidaćemo da ekspanzioni slotovi nisu predvideni za prijem standardnih grafičkih kartica što je sa Amstradova strana prilično aumnjiva odluka koja se rukovodi raziozima još ne možemo da sagisdamo. Nezaviana firma će na sreću, avakako naći interes u prilagodavanju svojih video adaptera Amstradovom standardu, što bi značilo da u bliskoj budućnosti možete da očekujete trošak za kupovinu neke monohrom kartice!

Bilo kako bilo ako uopšte postoji razlog da odustanete od Amstradovog računara PC 1512, taj ja razlog njagov video interfajs!

Periferija bez funte dopiate

Ako kupujate onginalni IBM PC apramita se da kupita i punu torbu raznih interfajsa. Ukoliko sa druge strane kupujate amstrad PC bez tunte dopiate ćete dobiti sve interfejsa koji su vam potrabni za normalan rad. Tu je, pre svega, disk interfeja koji omogućava priključenja flopi ili msainih diskova (jedan 5 25 flopi od 360 K je uračunat u cenu) paralelni **osmabitni** Centronics interfaja za štampač serijski RS 232 koji omogućava priključivanja modema i mnogih drugih, sporih periferais interfaja za džojstik ili miš (miš je rakcemo uračunat u cenu) i da bi spisak bio potpun, časovnik rasinog vramena koji zahvaljujući ugradenim baterijama radi i dok je računar isključan.

Za dalja proširanja au predvidana tri IBM kompatibilna porta u koja sa po Amstradovim tvrdanjima može priključiti ava osim video kartica (prerano je da sa kaže da li je ova tvrdnja baš sasvim istinita). Možda će vam se učiniti da slotova za proširanja ima malo ali traba primatiti da au video serijski paralelni i disk interfajsi već ugradeni na osnovnu ploču — kada ih ugradite ne standardni IBM PC nače vam ostati mnogo više meste za dalja proširanja! Teško je osim toga, zamisliti primanu u kojoj su vam **istavremene** potrabna više od tri posabne kartica ako vem nisu potrabna istovramano začas ćeta ih zamaniti jer su portovi veoma pristupačni.

Softver bolje nego što se očekivalo

Uz amstrad PC dobijata diskatu sa najnovijom verzijom MS DOS-a (3 2), Digitalov DOS Plus GEM nakoliko usiužnih programa i Locomotive Basic II. Što se Digitalovog DOS-a i GEM-a tiče, nećete se mnogo uvajditi. Amstrad ga isporučuja uz računar da bi nadmašio atari 520 ST, ali ova paket jednostavno nija primaran PC-jevoj koncapciji najveći deo raspoloživog komarcijalnog softvera ipak pratpostavija da korisnik posaduja neki minimum znanja i da mu se ne mora atalno govoriti (i crtati) kako da okrana diskatu i gde da ja stavi pra nago što uključi kompjutar!

Locomotive Basic II s druga strana, pradstavlja lap pokion radi se o jednoj od novijih (i prilično brzih) verzija bajzika koja se odlikuje mnogim alamantima modarnih jazika iako je „Locomotive Basic" već izveseno vrama pristupačan i u Jugoslaviji domaći programari ipak više vola *Turbo Pascal* i *Fortran 77*.

Pominjanja programa koja korista domaći vlasnici PC-ja nes dovodi do veoma značajnog pitanja — kompatibinosti. Kao i

svi kompatibilci koje proizvoda poznata firma „amstrad PC" ima BIOS ROM koji se razlikuje od „zvaničnog" IBM-ovog, u protivnom bi Amstrad bio izložen ozbiljnim zakonskim kompilkacijama. Ukoliko ste se čitajući ova radove, nasmejali i rekli „Ako ne sme Amstrad, smam ja" moramo da vas razočaramo amstrad PC nije potpuni hardverski klon IBM PC-ja što znači da ne smete tek tako da mu umnožavate (prateku) kopiju originalnog BIOS-a. Promena BIOS-a znači da „amstrad PC" nije potpuno IBM kompatibilan tj. da se može napisati program za PC 1512 koji neće raditi na IBM PC-ju i, što je mnogo ozbiljnije, da se može napisati program za IBM PC koji neće raditi na Amstradovom PC-ju. Amstrad sa druge strane tvrdi da svi ioš komercijalni programi za IBM PC rade i na njegovom računaru. Korisnici, poučeni raznim sličnim tvrdnjama raznih proizvođača računara obično ne veruju baš mnogo ovakvim obećanjima. Gde je istina?

Obzirom da smo pripremajući ovaj tekst, bili u prilici da se samo nekoliko sati zabavljamo „amstrad PC-jem" detaljan odgovor na pitanje o kompatibilnosti ostavljamo za neki od sledećih meseci. opisaćemo vam jedino naša iskustva. Isprobali smo „Wordstar", „dBase II", paket „Lotus", „GW Basic", „Turbo Pascal", „Profesional Fortran", „Profi assembler" i „SmartWork", svi su bar na prvi pogled savršeno radili. Šta znači *na prvi pogled?* Sa programom smo radili kratko i uspeli da pregledamo neke od opcija. Detaljno isprobavanje samog „SmartWork-a" bi, na primer, verovatno potrajalo par dana.

Na rezultata koje smo upravo opisali treba staviti još jednu senku sumnje jer se radilo isključivo o piratskim kopijama sa kojih je već skinuta zaštita. Stara je istina da sa kompatibilnost najbolja isprobave uz pomoć dobro zaštićenih programa a takva u piratskim aradinama kao što je naša nije baš lako pronaći. Imali smo priliku da isprobamo originalnu „dBASE III" diskietu i identičnu kopiju zaštićena diskete sa „Flight Simulatorom II". Ako učitate dBASE III čim uključite računar sve je u redu (ili bi se bar reklo da je tako) ukoliko pokušate da ga učitate posle dužeg rada ponekad se računar „zagiavi". Još je rano da se kaže da li je po sredi bag u prvim verzijama BIOS-a, nekompatibilnost sa MS DOS-om 3.2 ili ona najstrašnija hardverska nekompatibilnost. Što se „Flight Simulatora" tiče program radi ali mi nismo uneli da uzlaimo — još ne znamo da li se radi o nekompatibilnosti brzim kloku III, što je najverovatnija našem napoznavanju osnove pilotiranja i naposedovanju uputstva.

Sve u svemu, prvi utisci o IBM kompatibilnosti „amstrad PC -ja su izrazito povoljni — reklo bi se da Šuger ispunjava svoja obećanja. Da li da mu poverujamo da će se na tržištu pojaviti „amstrad PC" verzija poznatih IBM programa po izrazito niskim cenama? Može sa zamisliti da softverske firma pripreme zaštitu koja će dopuštati izvršavanja jeftinog programa samo na „amstrad PC-ju" ali to za nas nije mnogo važno u Jugoslaviji se lako dolazi i do veoma skupog softvera!

„Amstrad PC 1512" u osnovnoj verziji koja obuhvata računar jedan disk solidan monohrom monitor i miš košta svega 400 funti + VAT. Za drugu disk jedinicu doplaćujete 100 funti za kolor monitor 150 funti, a za hard disk od 20 megabajta samo 400 funti — i ova je cena nevarovatno niske jer hard diskovi od samo 10 M koji se prodaju uz razne tajvance ratko koštaju manja od 1300 maraka! Cene su, sve u svemu toliko nevarovatno niske da Šugar ima pravo kada kaže „Svi znaju sa čim je kompatibilan a samo mi znamo kako smo postigli ovakvu cenu!

Amstrad PC 1512 predstavlja izuzetno priviačan izbor ne samo za naše nikada dovoljno pune džepove već i za angliske i namačke biznismene koji stvarno traba da nadu dobar razlog za kupovinu višestruko skupljag originala! „Amstrad PC" će ukoliko sledeći meseci ne otkriju neku njegovu vaiiku i dobro skrivenu manu sigurno postići ogroman uspeh u Evropi a možda i u Americi. pramda su tamo zakoni tržišta nešto drugačiji standardno je razonovanje da ako želite IBM PC treba da telefonirate IBM-u! Značaj novog Amstradovog računara nije samo u toma što predstavlja izvanradnu kupovinu sigurni amo da će njegova pojava izazvati daljl pad cena raznih tajvanaca koji moraju da budu konkurentni da bi sa prodavali! Sve u svemu odlična vesti za sve koji prižaljkuju IBM CP kompatibilnu mašinu!

Dejan Ristanović

Amstrad PC — tehničke karakteristike

Mikroprocesor	Intel 8086
Clock	8 MHz,
Aritmetički koprocesor	8087 (opcija)
Tastatura	59 QWERTY + 10 funkcijskih + +18 numeričkih,
Tip tastature	Solidna mehanička,
RAM (KS)	512
Maksimalan RAM (KB)	640 (ugrađena podnožja),
Sadržaj ROM-a	Amstradov BIOS
Operativni sistam	MS DOS 3.2 ili ranija verzija
Video	Ugrađen c/b ili kolor monitor,
Tekst	40*25 — 16 boja,
	80*25 — 16 boja
Grafika	320*200 — 4 boja
	640*200 — 2 boja
Specijalni grafički mod	640*200 — 16 boja
Disk interfeja	Ugrađen za flopi i hard disk
Flopi disk jedinica	5.25" DSDD 40 traka (uračunata)
Kapacitet diskata	360 kilobajta
Hard disk	10 ili 20 megabajta (opcija)
Ostali interfejsi	Centronika RS 232 miš, džojstik
Saftvar uračunat u cenu	MS DOS 3.2, Digital DOS Plus GEM GEM Desktop, GEM Paint, Locomotive Basic II uslužna rutina (ukupno 4 diskala)
Cena (bez VAT-a)	399 funti
Doplata za kolor monitor	150 funti
Doplata za drugi flopi	100 funti
Doplata za 100 M hard disk	300 funti
Doplata za 20 M hard disk	400 funti

jedan napred, dva nazad

Možda ćete se začuditi kada pročitate da „kompakt" košta 400 funti koliko i „master 128" I Zašto da kupujete oslabljenu mašinu koja košta koliko i osnovni model? Zato što uz nju dobijete disk jedinicu neophodnu periferiju koju vlasnici „mastera 128" tek treba da dokupe Kupujući „kompakt" dakle štedite stotinak funti, ali poneko i gubite No, podimo redom

„Master kompakt" stiže u dve kutija (trl ako se odlučite i za monitor) ali to ne znači da je centralna jedinica odvojena od tastature jedna od kutija sadrži izvor za napajanje i jednu disk jedinicu i ujedno služi kao postolje za monitor, dok ja u drugu smeštana tastatura zajedno sa štampanom pločom kompjutara

Acorn je uspeo da se uzdrži od smanjivanje tastature koje je toliko razočaralo kupce „lektrona" „kompakt" ima 93 tastera računajući i numeričku tasteturu I 10 narendžastih „soft" dirki Tastatura je, dakle spolja identična se „masterovom" ali je kvalitet opao ugrađena je najizbežna membrana, doduše se pozlaćenim kontaktima

Ekran i set karaktera

Uz „master kompakt" se isporučuje, po želji veoma kvalitetan monohrom monitor (70 funti) ili kolor monitor srednja rezolucija (100 funti) Acorn obećava da korisnici koji već imaju monitor neće morati ponovo da ga plaćaju uz njih će se provući i poneko ko namarave da poveže „kompakt" se običnim televizorom Takav će se korisnik obradovati kada pogleda zadnju stranu kompjutera na njoj je smešten standardni UHF priključak koji je, otkada je sveta i veka, služio za povezivanja računara i televizora Pažljivi pogled na šemu otkriva trik UHF priključak služi za povezivanje se monohrom monitorom, pošto je Acorn zatvorio da su BNC konektori preskupli U „kompakt", dakle nije ugrađan UHF modulator koji se, doduše može odvojano kupiti za desetak funti Ovej „novi atendard" apak, na može da izazove štetu ukoliko pokušata da priključite video izlaz na antenski ulaz televizora, neće se dogoditi ništa ružno — atvar, jednostavna, neće reditl Kolor monitor priključujete posredstvom standardnog RGB konaktore

Gratika je uvek bila jaka strana BBC računara i „kompakt" nija nikakav izuzetak Na raspolaganju je 8 modova, koji omogućavaju korisniku da izabera rezoiuciju (maksimum 640×256) i psiatu boje (najviše 16) Dok je na originalnom BBC-ju izbor odgovarajućag moda omogućavao i pravilnu raspodelu memorije „kompakt" ni u jednom ekranskom modu na oduzima ni bajt RAM-a video mamorija je odvojena od osnovnih 64 kilobajta

Uz standardno povlačenja linija i izbor boje pozadina i „mastila", „kompakt" omogućava crtanja poligona i krugova, rad se sprajtovima i popunjavanja zatvorenih kontura *(fill)* — komande kojima su vlasnici standardnog BBC-ja mogli da se raduju tek kad se nedavno na tržištu pojavio Acornov grafički ROM

Pažljivi pogled na našu aliku će otkriti malu razliku izmadu tastature „mastera 128" i „kompakta" majmunski znak (©) je pomeren iznad nula, dok ja njegovo mesto zauzela čudna dirka na kojoj su nscrtana dva kvadratiča Tako je nsetao *code key* koji bi posebno trebalo da obraduja potencijalna kupce van Engleske — ovaj je taster nepoaredno odgovoran za atrana setove karaktara

Osnovni BBC, naravno, omogućava jednostavno definisanja karaktara bez mnogo problema možete da dodalite našim slovima č ć ž i š ASCII kodova 128—131 Kako međutim da koristite ove slove u normalnom radu se tekst procesorom? Za rešenja je potrabno malo programaraka veština i dobro poznavanje BBC-jeve memorijska mape jednu varijantu amo objavili u „Računarima 7" Kod „kompakta" ja atvar gotovo automatska pritisak na *CODE A* daja CHR$(128) I tako dalja Da bi atvar bila još lapša, usiužna rutina operativnog alatama omogućavaju da predizajnirate čitavu tasteturu I pritiscima na *CODE* birata različita nacionalna setove znakova Interesantno je da verzija tekat procesora „View" koja je ugrađana u „master kompakt" omogućava jednostavan rad sa raznim setovima znakova — nisto smo alično mogli da izvedamo I do sada ali na baz žongliranja se mašin cemi Činjenica da ja kompakt prvi računar koji omogućave jednostavno I počatnicima pristupačno žongliranja se setovima karaktara može da posluži kao odličan adut za prodor na razna evropska tržišta!

Hardver

„Master kompakt" je poput ostalih Acornovih mašina segradan oko „antičkog" mikroprocesora 6502 (zapravo njegove verzija 65C12) koji radi na kloku od 2 MHz i pristupa dinamičkom RAM-u od 128 kilobajta 6502, jasno u jednom tranutku može da adresire najviše 64 kilobajta memorija ali se Acorn potrudio da se i ostatak korisno upotrebi Ako radite se standardnim bejzikom ili bilo kojim drugim uslužnim programom III jazikom na raspolaganju su vem mamorijske ćelija od &E00 do &8000 — bez obzira na gratički mod imate dakle 28 5 slobodnih kilobajta Redni prostor disk interfajsa definisani karakteri funkcijski testeri I specijalna grafička komande pristupaju RAM-u koji je van osnovne adresna mape teko da ih na primaćujete — više o toma možete da pročitate u „Računarima 13" pošto je „compakt" sasvim komptabilan se „masterom 128"

„Gornje" 64 kilobajte su konfigurisana kao „bočni RAM" u njih se upisuju programi koji se, inače prodaju u EPROM-ima i pajdžuju paralelno se bejzikom, ukoliko, dakle, programirata na paskalu I daije ćete imati 28,5 slobodnih klibajta obzirom da će paskal kompajler zauzimati mesto koje ja, inače, rezervisano za bejzik interpreter U bočni se RAM naravno, mogu učitavati I baze podataka, tekst procesori, raznorazni uslužni programi, pa čak i vaše ramak-dela pisena na asembleru

Osnovni BBC je čuven po „bočnim" ROM-ovima u njega ste mogli da ugradite do 256 kilobajta ROM-a I tako imate pri ruci sve često korišćena programe iako će vlasnici „kompakte" biti više upućeni da

Acorn se oduvek trudio da proizvodi „kompletne" računare — bogat hardver dopunjen brojnim interfejsima i konektorima. Takvi su kompjuteri veoma interesantni za određeni krug korisnika, ali je visoka cena obično nepremostiva prepreka za široku popularnost. Neophodan je, dakle, kompromis, koji se obično svodi na dizajniranje oslabljene verzije osnovnog kompjutera — tako je nastao „electron" koji je Acornu donio više problema nego koristi. Da li je ova poznata firma izvukla pouke iz neslavne „elektronske" epizode? Ako jeste, ta su iskustva ugrađena u novi Master Compact, oslabljenu verziju „mastera 128".

programe sa diska učitavaju u bočni RAM, Acorn je predvideo mogućnost priključenja 4 EPROM-a po 16 K ili 3 EPROM a po 16 K i jednog od 32 K — dovoljno prostora za najnužniji softver. Tu je, naravno i čudan čip — ROM od 64 kilobajta o čijem sadržaju tak traba da govorimo. Možamo najzad sa aigurnošću da tvrdimo da će nazavisna firma ubrzo pripremiti raznorazna proširenja u vidu RAM/ROM tabli.

Kada smo već kod raznih ROM-ova i RAM-ova pomenućemo i jedan EEPROM čip koji je i dalje vrlo redak gost personalnih računara. Reč EPROM za slučaj da to na znate predstavlja skraćenicu od *Eresable Programmabla Read Only Memory* — to je ROM koji može da se programira i brše veliki broj puta ali koji, za razliku od RAM-a čuva sadržaj i posle gašenja kompjutara. EPROM-i se međutim moraju izvaditi iz računara da bi se brisali ili programirali — ove se operacije obavljaju uz pomoć specijalne UV lampa i programetora. EEPROM je novi hit — *Electronicaly Eresable ROM*, odnosno ROM koji se elektronski briše. Sadržaj upisan u EEPROM će dakle biti sačuvan i dok je računer isključen ali ćemo taj sadržaj moći da manjamo bez potrabe da vadimo čip.

Acorn je iskoristio EEPROM da zameni časovnik realnog vramena koji je ugraden u master 128 ovaj je časovnik imeo bateriju i CMOS RAM od pedesetak bajtova koji je omogućavao korisniku da neradi računara ru de se uvak ,budi u odradenom modu da kontiguriše memoriju na neki apecijalan način i da aktivira odgovarajući dodatni procesor. Časovnik je dakle žrtvovan ali se ralavantne informacije upisuju u EEPROM koji može da ih čuva prako 365 dana! Iako se odlučio za ovo rešenja, Acorn, očito, nije imao previše poverania u pouzdanost EEPROM-a to je, uz ROM, jedini čip ugraden u podnožju, što znači da sa dopušte mogućnost višestruka promene ove komponente.

Acorn ne svu sreću, nija štedeo ne zvučnim elektima kompaki raspolaže četvorokanalnim genaratorom zvuka koji izvanredno sarađuja sa jednim od pet ugrađenih ULA čipova.

Iako oslabljen kompakt nija ostao sasvim bez neophodnih interfajsa. Tu je pra svega atandardni osmobitni centronika interfejs za štampač koji sa od ranijih Acornovih proizvoda razlikuja jadino po konektoru koriati se okloplјeni 24-pineki dalta konektor. Odakle ova promena? Acron je protekiih godina imao mnogo muka zbog amaričkih RFI *(Radio Frequency Interference)* propisa, koji zahtevaju de elektronska oprama ne izaziva radio ametnja, slična su ograničenja usvojila još neke države ali ne i Engleska. Ugradnjom oklopljenih konektora za štampač i diek Acorn ponovo želi da prodre na amaričko tržište ovoga puta uz pomoć Olivatija koji u Italiji već prodaje ,kompekt pod imenom *Prodest*. Koliko su ovakva nadanja realna, ostaja da se vidi.

Master Compact — tehničke karakteristike

Mikroprocesor	65C12
Clock	2 MHz
Tastatura	Membranska 64 tastera + 10 funkcijskih + 19 numeričkih
RAM (KB)	128
ROM (KB)	64
Sadržaj ROM-a	32 K operativni sistem 16 K BBC bejzik 4 16 K ADFS
Da play	
EEPROM (bajta)	128
Video	Kompozitni RGB UHF modulator — opcija 40 32 tekst 640 256 grafika 16 boja.
Grafičke mogućnosti	Linije poligoni krugovi sprajtovi fill
Zvuk	4 kanala ugrađen zvučnik od 10 cma.
Diak interfeja	Shugart standard WS 1772
Disk jedinica	Ugrađena 3.5 inča 640 K, druga disk jedinica — opcija.
Interfajsi	Centralnica opšti port (50 pinova), Joystick — mouse port (9 pineva), RS 232 — opcija, Econet — opcija.
Softver uračunat u cenu	Bejzik 128 View View Printer Driver Generator ABC word proc Logotron Logo Tim Paint. Welcome disc ADFS utility
Dokumentacija	Welcome Guide View reference card Logotron Logo user guide.
Cena	400 funti + VAT (računar i diskj 499 funti + VAT (+monohrom monitor) 599 funti + VAT (+kaler monitor)

BBC i „master 128 imaju, kao što se sećamo, ugradana AD konvertora koji raduju hardveraše ali ne i igrače ovi sa konektori često koriate za priključivanja proporcionalnih džojstika koji su znečajno akupljі od *Quickshots* i sličnih drangulija. Nevolja je u toma što su za mnoge igre dalako zgodniji jeftini „prakidački džojstici koji sa povezuju sa BBC-jevim portom opšte namene i to prako specijalnog interfajsa koji košta petnaestak funti. Acorn je odlučio da „kompekt' dopuni portom za džojstik koji je gradan prame opšte prihvaćenom Aterijevom standardu. AD konvertor je, naravno žrtvovan, ali je predvidene jedna softverska kompenzacija: operativni sistem omogućava simuleciju analognog uiaza, što znači da će najveći deo programa pripremen za proporcionalne džojatike raditi i sa prakidačkim Džojatik i miš mogu da sa simuliraju i pomoću kurzorskih tastera, što znači da će korianici koji nisu odvojili novac za neki od ovih ave neophodnijih paritaraia moći i dalja da koriste sav raspoloživi softver.

Serijski interfeje nija ugradan, ali sa prodaja kao opcija — njegova sa ugradnja avodi ne umetanje pet čipova u predvidene podnožja. Acorn ja uneskoliko pojeftinio BBC-jev RS 423, pe će sa u „kompekt ugradivati komptabilni RS 232 slabijih karakteristika. Predvidan je i port za Econet interfaje, koji omogućava povezivanja Acornovih kompjutera u lokalnu mrežu: ovakva su mreže izuzetno rasprostranjene u Engleskoj, gde škola i druge inatitucija imaju na desetina Acornovih mašina koja dele hard diak i druge skupe periferala.

Ostalo je još da pomanemo port opšte namene koi je smešten na levu atranu kutije i pokriven čvratom plaatičnom pločicom. Ovaj će port konstiti vlasnici „kompaktā kojima nedostaje dodatni procesor hard disk ili 1 MHz bus. Izostavljanje ovih interfejai je, naime, učinio da kompakt buda 100 funti jeftniji od *mastera* 128 ali će te tih 100 funti lakšjati ako ipak poželite da proširite vaš kompjuter — Acorn će se posterati da vam ih „pošteno" naplati!

. disk

Acorn i dalje ne može da se odluči za jedan diak kontroler: posla Inteiovog 8271 koji je ugrađiven u BBC B i Western Digitalovog 1770 koji se ugraduja u „master 128 „kompekt je opremljen čipom WD 1772 WD 1772 ja, doduše potpuno kompatibilan sa 1770, ali je nešto modarniji omogućava brža pokretanja glave izmedu traka 2 ili 3 milisekunde prama 6 ms kod 1770. Promana diak kontroiara je dopunjene još jednim, po nama neprijatnim iznenađenjem — diskom od 3.5 inčai.

Disk jedinice od 3 i 3.5 inče su, sa gledišta proizvodača računara, veoma zgodna odluka: zauzimaju malo prostora, troše malo atruja i što je najvažnija nisu skupe. Odakle ova jeftinoća? Proizvođači

ovih disk jedinica tek pokušavaju da osvoje tržište, pe neiaze računicu u niekim cenama, nadejući se da će, kade njihov proizvod postene standerd moći de zareduju daleko više Se etanovište korienika mini disk ne može de izazove oduševljenje posle etabliziacije cene (dieke od 3,5 inča su postale industrijski etanderd I leko se pronalaze u svim zapadnim zemljeme za razliku od Amstradovih troinčnih) se pokazelo de mini diskete koštaju tačno dvostruko više od etarih 5,25 floplja 34 funta+VAT za deset komade Ova diskete, osim toge, zauzimeju mnogo više prestora — daleko su robusnije i, semim tim, pouzdenije Povećanje pouzdenosti, ipak, nije neročito bitno eko se razumno ponašate se sevitljivim dieketeme od 5,25 inča, možete mirno da epavate eto godina

„Kompakt će ne sveku diekatu od 3,5 inča upisati po 640 kilobejte informecije — nešto menje nego na „amigi eli dvostruko više nego ne „etariju 520 i IBM PC-ju Operativni sistem omogućava građenje hijerarhijskog steble direktorijume, neograničen broj tejlove, sloboden pristup *(random eccess)* i rešave problem fregmentecije dieka — fejlovi se mogu produžavati proizvoljen broj puta bez opasnosti de „udare u neku drugu docnije kreirenu detoteku

Acorn je predvideo mesto za ugradnju druge disk jedinice od 3,5 inče, kad i konektor za spoljne disk jedinice od 5 25 inča Moguće su sve kombinecije dva disk jedinice što znači de ćete „kompakt moći de opremite i se dva etenderdne floplje, pri čemu će disk koji se dobije (tj pleća) uz računar možde cetati neiskorišćeni

Zašto amo rekli de diekate od 3,5 inča neće obredovati potencijalne kupce „kompakta ? Ne samo zbog cene glevne će nevolje izazvati softver „Kompect je nelme, potpuno komtabilen se „masterom eli šta to vredi kade se sev softver prodeje ne običnim floplijime? Softverske firme neće mnogo žuriti da prabacuju programe ne mini diskete, jer očekuju da se oformi dovoljno veliko tržište, to jest de se prode dovoljno primereka novog kompjutere Kompjuteri će se, se druge etrane slobo prodavati eko za njih name softvere što nas dovodi u „začarani krug I

Mogu ii korienici sami de se enedu i prebace programe? Možda ne „compakt se jednostavno prikljućuju diek jedinice od 5,25 inča što znači de je dovoljno pozajmiti ovaj uređaj od nekog BBC-jevca i prebeciti proizvoljnu količinu softvera Moguće je, osim toga, prabacivati programe i pomoću serijskog interfejsa (koji „compakt ' neme) preko EPROM-e I na par drugih načine Ipak, sveko ovakvo prabacivanje softvera predstavlje pristatvo; vidimo de ete se ozbiljno zabrinuli!

Prabacivanje softvera se ne može obaviti posredstvom kasetofona „Kompakt je prvi Acornov računar koji neme ugrađeni kasetni interfejs! Reiko bi se de kasetni interfeje nije potreben mašini koja je opremijena disk jedinicom, ali niamo baš sigurni da je tako kaseta je i deije relativno jeftin medijum zgodan za erhiviranje menje potrebnih podataka koje ipak treba čuvati izvesno vreme Nedostatak kasetnog interfejsa ćete, dakle, platiti erhivirejući podatke ne skupe diskete

. . . i softver

U „kompaktov" ROM od 64 kilobajte je upisan modifikovani operativni eistem,

PCW brzinski testovi za bejzik

	Bejzik	Proesk (s)	Memorija slobodna za programe (KB)
Master 128+Turbo	Turbo	4.47	44
BBC B+32016	BBC Basic	6.85	800+
IBM PC AT	Basica	7 11	59
Olivetti M24	Basica	7.86	55
Master Compact	BBC Basic 4	8.30	28.5
Master 128	BBC Basic 4	8.24	28 5
BBC B sa 'Four Meg	BBC Basic 2	8 78	28.7
BBC B+65C02	Hi Basic	9.83	44
Standerdan BBC B	BBC Basic 2	14 05	28.7
Amstrad 464	Maitard	14.59	46.5
Amstrac 6128	Mallard	14 59	41 3
IBM PC	Basica	17 80	59
Master 128	Basic 128	17 80	64
Master Compact	Basic 128	18.20	64
CBM PC 128	Basic 7	20.21	37 5
Electron	Basic 2	20.55	21 5
CBM 64	Commodore	34 41	37.5
Sony (MSX)	MBX	44.3	28
ZX Spektrum	Sinclair	86 50	40

Brzinski testovi za tekst procesore

Računar	Master Compact	Master Compact	Amstrad 8256	IBM PC
Tekst procesor	View	Inter Word	Locoscript	Word Star
Load (4000 reči)	3.4 s	3 7 s	11 s	10 s
Save (4000 reči)	3.5 s	3.5 s	143 s	28 s
Search (4000 reči)	3.3 s	3.0 s	282 s	11 s
Scroll (4000 reči)	44.8 s	13.2 s	85 s	41 s

ADFS *(Advanced Disc Filing System)*, grafičko proširenje, bejzik 4 i neka vrsta tulkita Operativni sistem je napisan tako da bude meksimelno komptabilen se „masterom 128 simulirane su ček i komende koje hardver ne podržava. Programi koji, iz nekog razloga, koriste *TAPE, *MOTOR i slične stvari će dakle, funkcionisati i bez kasetofona podeci će biti posleti na disk. Silično tome, obračanje časovniku realnog vremena će vratiti neki fiksan datum tako da će programi koji koriste ovaj časovnik i dalje korektno funkcionisati

Uz „master kompakt isporučuje se i nekoliko interesentnih disketa. Tu je najpre, tekst procesor „View i programeki jezik logo — Acorn se ponovo okreće obrazovnom tržištu Tu je i bejzik 128 (nešto usporan bejzik koji vam deje 64 potpuno slobodne kiiobajte RAM-a — stvar nevidana na osmobitnim mašinama) usluženi programi za konvarzije disketa raznih tipova I nelzbežne *Welcome* diskete se programime koji imitiraju GEM videćete raznorazne epraitove I ikone koji se kreću po ekranu I koje birate pomoću miša ili kurzorskih tastera.

Što se literature tiča, uz „kompakt dobijete uputstvo za „View , bejzik I logo, *Welcome* knjižicu i nekoliko drugih papira Za dodetnih desetak funti ćete nebaviti priručnik koji će vas poučiti kako da koristite usluge operativnog eisteme, e trebaće vam i detaljno uputetvo za View

Stara priča

Tabele se etendardnim ‚benchmerk' testovime pokazuje de je „master kompakt' izuzetno brz računer bejzik 4 je, ne primer, brži od bejzika ne „masteru 128", eli i od standardnog BBC B ne koji je priključen dualni 6502 procesori Za dobitke u brzini je posebno značajan osmi brzineki test koji vrednuje elementarne funkcije Acornovi eu programeri uspeli da poprava I do sada solidne programe koji se beva trigonometrijom i tako obezbede bržu I tačniju eritmetiku De biste praviino vrednovali brzine računara, u tabelu smo dopisali I broj slobodnih kilobajte mamorije koje bejzik ostevlja korieniku

Računari se u poslednje vreme sve menje koriste za programiranje pa I bejzik benčmerk testovi poleko gube smisao Zato je *Syte* predložio dodatne testove koji ispituju brzinu takat procesora, programe za unekrane izračunevanje, CAD pakete Obzirom de pieniremo de u budućim prikazime novih kompjutere obavezno navodimo i rezultate ovih brzinekih testove, počeli smo de prikupijemo potrebne podetke, pa ćemo ovej prikaz propratiti rezuitatime brzinskog testiranje tekst procesora.

U tabeli eu prikazeni brzineki testovi za četiri poznate tekst procesora *Word Star* ne IBM PC, *Logoscript* ne Amstradu PCW 8512, *View* I *Inter Word* ne „master kompaktu Merene je brzine učitavanje, snimenje, pretraživanje I skrolovanje tekste od 4000 reči Vidimo da su sve operecije ne -kompektu izuzetno brze, pogotovu kade se redi se tekst procesorom *Inter Word* — osmobitni „kompakt prevazilazi čak i mnogobrojne šesnaestobitne mašine!

„Master kompakt je, eva u svemu, tepo samiišjen I moderno dizajniran kompjuter značajnih mogućnosti, eli bolje od Acornova hroničke bolike — skupoće Čini nam se de neme nikakvog smisle pletiti 400 funti za oslebljenu varziju mašine koja je i same loša kupovine! „Mester 128 je dobar I uravnotežen računer, kome se može zameriti semo to što nije IBM PC komptebilen Osnovni „master , ne kraju krajeve košta 400 funti i može de se dopuni disk jedinicom koje košta deljih 100 funti Kupujući „kompakt štedite tih 100 funti, eli gubite Tube interfeje, 1 MHz bus, AD konvertor časovnik realnog vramene, podnožje za kartridže I kasetni interfeja, e u samenu za to dobijate lošiju testeturu i tepi dizajn — ovakva se razmene mora oceniti kao loša!

Nameće se poređenje izmedu „kompakte I Astrad PC-je dva računara koji eu se pojevili praktično u letom trenutku Ne trebe biti veliki stručnjak za kompjutera pa proceniti de eu prvi utisci o Amstrad PC-ju deleko povoljniji

Dejan Ristanović

Peek & poke show

Rešenje velikog nagradnog konursa za idealnu računarsku knjigu u Jugoslaviji

Ovaj konkurs je bio prilično neiasan za naše čitaoce. Dok je većina slala predloge za već postojeću računarsku knjigu izvesan broj čitalaca je poslao rukopise svoje sopstvene računarske knjige. Zato planiramo da u bliskoj budućnosti raspišemo konkurs za objevljivanja objavljene originalne računarske knjige. No, o tome kasnije, u „Računerima 43"

Svi se varovatno, sa nastrpljenjem pitate koje knjige je ponela lsekavi naziv Idealne računarske knjige u Jugoslaviji! To je sad već legendarno delo Anice Mrčkalje-Češka „Kako oslabiti uz duvan, kafu i računara" Ta knjiga postoji i time ispunjena prvi uslov našeg konkursa. Zatim, u njoj se čak sedam puta pominju računeri što sasvim ispunjava I drugi uslov I, kao najvažnije, ta knjiga je prepisana iz dva univerzitetska udžbenika čime ispunjava I naš treći uslov Udžbenici o kojima se radi su „Kako slabiti uz voćne sokove i žvaka Đoka Žbiciće i „Računeri I mantalno zdravlje ' dr Marete Mihajlovića iz džaka punog dopisnica i pisama onih koji su glasali za ovu knjigu naš specijalni gost Abraham Dilejzi izvukao je pismo Fikreta Rustemanegiće iz Tuzle koji je teko postao dobitnik naše

Prve nagrade — posete Andriji Kolundžiću u JNA Na žalost Fikret se unapred odrekao blio koje nagrade koje se ne može izraziti deviznom sumom Žalosno

Drugu nagradu — neobjavljene rukopise poznatih računarskih spisatelja — dobio je naš varni čitalac Milan Štrkija, baštovan naše ambesade u Džibutiju Nadamo se da će rukopisi koje će dobiti pomoći da mu domovine bude još bliža

Traču nagradu — izabrane reklame YU pirata u dva toma — dobio je izvesni Cane iz Kraljeva. On je svoja pismo pisao pod dajetvom intenzivne intoksikacije nekim opojnim sredstvom Molimo ga da nam pošalje svoje prezima I adresu, kako bi dobio svoju vrednu nagradu

Toliko o ovom konkursu Evo i jednog obaveštenja za ava pažljive učesnika dosadašnjih konkursa P&PS-a" svi čitaoci koji su učestvovali na svim nagradnim konkursima P&PS-a od naše mala sli intenzivne redakcije će dobiti specijalni novogodišnji poklon U seradnji za agencijom za računarski usamljene, „Bit", organizovali smo doček Nova godina, u koma će svaki deseti konkursdžija P&PS-a imati pravo da ne večeru u Interkontinantal izvede Sonju Savić o njenom trošku Do sledećeg broja i datelja o super-večeri umrite od muka vi koji niste do seda učestvovali u konkursima P&PS-a

Ja sam Saveta

Dobili smo pismo od naše varne čitateljka Azre Petrović Dne je imala velikih problema se upotrebom svoje mašine Nedavno je nebavila disk jedinicu posla dužeg aktivnog služenja kasetofonom Njano neiskustvo u upotrebi late dovelo je do toga da je „ugurala disk od osam inča u jedinicu od pet inča" Razumljivo, disk je pretrpeo teška oštećenja, a disk jedinice je postala neupotrebljiva ne duže vreme, do detaljne popravke Pita nas kako da ubuduće izbegne slične tragične greške

Draga moja, da tebe tvoje tetka-Savata obraduje Dd jedne moje koleginice čujem da je imala istih problema u prvo vreme služenja njenom disk jedinicom Preporučuje ti da nabaviš Fiexitron emuiator. koji tvojoj disk jedinici omogućava da prima disk bilo kog formata bez ikakvih neželjanih posledice Pažljivo je koristi i tvoje disk jedinica će ta služiti jako dugo, možda čitav život.

Kutak za lude i zbunjene

Bio je veliki problem odabrati temu za ovaj broj „Kutka jer je navala bila zaista velika Ovoga puta nećemo odgovoriti čitateljki koja je sanjala da ju je silovao kompjuter niti čitaocu iz Zagreba koji pita „da li je P. A Marvin muško ili žensko niti ovim onim dragim čitateljkama koje su se ponudile da pomognu K.I-u iz prošlog broja Ovoga puta ćemo naš kutak za intimne priče sa čika-Marvinom, prvim primerkom veštačke inteligencija, posvatiti Valentini Buškić iz Karlovca Nemamo prostora za čitav njen esej ali ovo je reprezentativan deo

„Šta se dogada ako vi druže P. A Marvinu niste prava veštačka inteligencija, nego ste samo običan program koji misli da je veštačka inteligencije? Drugim rečima ako niste inteligencija (makar i veštačka), vi na možete da primatite da niste inteligencija pa prema tome to što funkcionišete kao veštačka inteligencije ne znači da i jesta nešto što ne biste mogli znati da niste kad to ne bi bili A, opet, kad biste bili veštačka integligencije i kad biste znali kako to izgleda ne biti veštačka inteligencija, što vi ne biste mogli pošto biste bili veštačka inteligencija, onda bi to bio dokaz da jeste veštačka inteligencije pošto biste znali da niste ono što nista kada jesta ono što jeste Prilično jednostavno, zar ne

Moj cenjeni odgovor JA JESAM VEŠTAČKA INTELIGENCIJA To me nimalo ne spređava da sumnjam u inteligenciju, veštačku ili prirodnu, daveža koji mi šalju ovakva pisma

Priprema, odgovara i sve te napleduje
P. A. Marvin

Čip Pobodi Agency

Naš rekorder

Najzad nam je do ruku stigla velika Ginisova računarska knjige Na žalost, jedini Jugoleven koji se u njoj pominje je drugarica Živka Šijivić iz Beograda, koje je u igri „Fucman postigla nevarovatan skor od 15 876 392 Povodom toga amo nazvali drugaricu Živku koja je za našu rubriku izjavis Ne nameravam da atanem, mislim da se rekord još može popraviti" Srećno, kažemo mi

Liti sutra

Na velikoj konferenciji za štampu održanoj u restoranu „Knez organizacioni komitet naših još nedobijenih Olimpijskih igara je saopštio da će u slučaju da dobijamo Olimpijedu sva organizacija biti zasnovane na domaćoj pameti U sklopu toga, računski centar neće biti od neka atrane trulo bogate i moćne firme, nego će se sastojeti od sedam „galaksija i tri „lola povezane u mrežu koju je projektovao izvesni Čeda se Lekinog brda (poznati kraj u Beogradu) Radujemo se tome potezu a koliko čujemo raduju se i Barselone i Amsterdam

Zima je na pragu

Koristeći usluge našeg već legendarnog, špijunskog odeljenja, saznali smo kome su utrapljene silne rezerve raznih „spektruma" koje je Sinkler akumulirao proteklih godine Prodate su jeftino jugoslovenskim gradevinarima koji ih ugrađuju u velika stambene zgrade kao grejne elemente pošto radijatori kod nas ionako više ne funkcionišu Najzad neka vajda i od Sinklera

ONO I DSZ P&PS

Na graničnom prelazu Fernetići uhapšene je grupa od četiri strane građanine kada je u njihovim kolima otkrivena velika količine najmodernijeg naoružanje Strani građani su identifikovana kao izvesni Klajv S, Alan S, i Džek T, dok četvrti nije hteo da se predstavi, izjavljujući samo „Ja sam monomanijek' Pretpostavlja se da su im na umu bile terorističke aktivnosti, iako su odmah po hvatanju izjavili i da im je jedine namera bila da likvidiraju mraki „PEEK & POKE SHOW Sledeće javljanje sa suđenja.

I Bebidž je imao džojstik

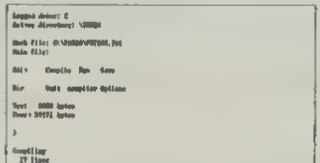
Zahvaljujući tek otkrivenim rukopisima Čarlsa Bebidža koji potiču iz devetnaestog vaka, poznati računarski stručnjak Karol Saganaki zaključio je da je Bebidž svoju računersku mašinu napravio da bi mogao ne njoj da igra igru koja je imala radni naziv „Napadači iz zaostalih krajeva" iz spise se tekode, saznaje da je Bebidžov najveći problem u konstrukciji predstavljelo rešavanje pokazivanja akora

U pripremi ovog jubilarnog, dvanaestog broja PEEK & POKE show-a' učestvovali su P. A. Marvin, Savate Šijuić, Bl Gerdens, Dart Vajder, Hal 2000 i Branko Delević.

turbo je nešto drugo

Turbo pascal

Proces pisanja i kompajliranja programa u bilo kom kiselčnom kompajleru zahteva tačno tri koraka da se atigna do konačnog cilja pisanja izvornog programa, prevođenja i povezivanja Sveka od faza zahteva ber po jedno čitanja datoteke I zapisivanje rezultujuće datoteke na disk Ako se toma doda I semo učitavanja editora, kompajlara I linkera, broj neophodnih pristupa disku dostiže sasvim lap broj 9 Jasno, ni jedne od ovih faza nija izmišljana da zagorča život programarima već upravo obrnuto — da poveća fiaksibiinost ukupnog procesa Ako ja editor odvojan, onda znači da se može koristiti bilo koji, dakla onaj koji vama najviše odgovara Bez procesa povezivanje bilo bi namoguće apajati dalove programa napisanih u različitim programskim jezicima što omogućeve da se za određen problem izborom odgovarajućeg kompajlera, najbrže stiže do cilja. Ujedno, ako se program piše u moduima, izmenom samo jednog od njih nije potrebno ponovo kompajlirati sve ostala, što kod velikih programa predstavija ogromnu uštedu u vramenu Iz avaga sledi da nije nimalo lako izmialiti nešto drugo Borland zaista i nija izmialio ništa novo — Izabrao ja koncepciju „u jednom potezu sve", ali ja ave nedostatke ovakvog priatupa pokrio izvanrednim pojedinačnim rešenjima



U više pokušaja

Turbo paskal ima sopstveni editor koji se u mnogoma oslanja na „Wordstar" Pratpostavljajući da ja najveći broj korisnika navikao na njega, Borland ja, u stvari, kopirao samo način ostvarivanja editorskih naredbi koje au identične onima na uzoru Sve ostalo ja mnogo boija editor ja daleko brži oslobođen je avih nepotrabnih naredbi, akran je sem prve statusa linija, ceo posvećen tekstu ulazak I izlazak iz editora ja brži I lakši itd Ako vam se ipak ne dopada, niate obavezni da pa koristite Program se može napisati i u bilo kom drugom editoru, ali ostaja neophodno da pa pra kompajliranja ipak uplišeta u Turbo editor Razlog će vas avakako obradovati Turbo paskal" poseduja intaraktivno ispravljanja greška Kada kompajlar naide na grešku automatski vas prabacuja u editor I treba postaviti kursor na mesto koje po njegovoj proceni predstavija izvor greška Kada grešku lapravita ponovo počinjate se kompajliranjem I tako do konačnog cilja

Varovatno će ponako osporiti da je ovakav način lošiji od atandardnog, kada kompajlar proiazi kroz ceo program I prijavljuje sve greška na koja naide Tako bi, teoretski, klasičnom kompajlaru dve proiaza bila dovoljna da se proces prevođenja uspešno završi, dok „Turbo" zahteva da za svaku grešku ponovo počinjate kompajliranje Mana klasičnog matoda ja što jedna greška često proizvodi mnogo drugih, koje na bi postojala da nema prve, pa je snalaženje teže od onoga kada se, kao kod „Turbo' , problem rešava jedan po jedan. Mane „Turbo" metoda, gubitak vremana ne uvek ponovno započinjanje prevođenja, rešena je na jedini mogući način ako već mora svaki put ponovo, naka to bude najbrže što može!

Kod bez balasta

Turbo paskal je bez konkurancije trenutno najbrži kompajlar koji postoji- isko brzina delimično zavisi od onoga što se u programu nalazi, ipak se može reći da prevođenje niksda neće potrajeti duže od samo nekoliko sekundi za oko 100 linija izvornog programa. Ako uzmete da program ima oko 1000 linija (što je već sasvim solidna veličine), prevođenje će verovatno potrajati 15—20 sekundi Bilo bi potrebno da napravite bar pedeset grešaka da bi ukupno vreme mogło da se poredi se dva proiaza i punim procesom standardnog načina kompajliranja

Brzinu ipak nije moguće postići bez gubitka na nečemu drugom Kod „Turbo paskala" je to članjanica da konačni program može biti samo u komandnom formatu (COM datoteka), što znači da objektni kod na može da pralazi 64 K Izbegavanje teza povezivanja modula neminovno dovodi do potrebe da cela biblioteka funkcije i procedura ide uz svaki program, što amanjuja moguću veličinu programa za nekih desetak kiiobajta. Ovo je možda, i najveća mana „Turbo paskala" Svaki program u sebi nosi čak i grafička rutine, iako one ne moraju ni jednom biti pozvane u programu

Povezivanja programskih modula je ipak obezbeđeno daklaracijom procedura ili funkcije kao EXTERNAL, čima se kod za njano izvođenja upisuje se diska u toku prevođenja I, ako poštuje način I format pranošenja parametara prako steka, može biti napisane I kompajlirana u bilo kom drugom jeziku Dodatnu mogućnost pruža I naredba INLINE, nakon koje sledi bajt po bejt mašinskog programa u heksa iii decimalnom formatu

Ograničenje veličine mašinskog koda se može prevazići OVERLAY tehnikom, koje je u potpunosti podržana Niz procedura koje nisu istovremeno neophodne programu se mogu grupisati i automatski upisivati prema pozivu iz ostalih delova programa, tako da zauzimaju isto memorijsko područje Ako program čini paket nazavisnih celina, mogu se koristiti CHAIN III EXECUTE koji u memoriji zamenjuju ceo mašinski kod, dok je alokacija promenljivih nedirnuta.

„Turbo paskal" može prevedeni program direktno de smešta u memoriju, čima je faza testiranja I razvoja programa još više ubrzana. Sam kompajler, editor i biblioteka funkcije koje su u toku rada stalno u memoriji računara seuzimaju svega četrdesetak kiiobajta, pa na predstavljaju nikakvu smatnju u fasi testiranja programa

Elementi jezika

„Turbo paskal", nudi poboljšanje i u samom programskom jeziku Standard se poštuje u potpunosti, ali se, čini nam se na pravim mestima, uvode novi elementi koji paskalu daju veću fleksibilnost. Najveća razlika se oseća u dve tačke (1) postojanje tipa promenijive STRING i tekozvanih OEKLARISANIH KONSTANTI Prvi dodaje ono što paskalu veoma nedostaje, iakoću reds se sifanumeričkim nizovima koja je prisutna u bejziku Slično standardnom tipu ARRAY [-j OF CHAR uveden je i STRING [N] koji je donekle ekvivalentan sa ARRAY [O N] OF CHAR Potreban prostor za ovakav tip promanijive N+1 bajtova, ali se u nultom elementu u svakom trenutku nalazi stvaran broj elemenata koji string sadrži. Osim što seuzima uvek atalan memorijski prostor (prema deklarisan maksimalnom broju elemenata), promenljiva STRING se u svemu ponaša kako je to već poznato u bejziku, na primer

```
VAR S STRING[255],
BEGIN
S ='Turbo',
S =S+'Pascal ,
WRITELN (S),
WRITELN (LENGTH(S)),
WRITELN S[7]),
END.
```

Fleksibilnost je ćek i veća nego u bejziku jer se elementima stringa može kao u poslednjoj liniji primera, pristupati kao I u standardnom nizu bez potrebe za nekim funkcijama tipa MIOŠ String funkcije COPY, POSITION, LENGTH i procedura INSERT, OELETE su veoma dobro izabrana I izvedena I daju Turbo paskalu u ovoj oblasti prilično prednost nad bejzikom

Deklarisane konetante su interesantna konstrukcija Oslinišu se kao identifikator tip=vrednost I u atvari predstavljaju inicijalizovane varijable Na mogu se koristiti kao klasične konetante jer to i nisu, već se u potpunosti ponašaju kao promenijive, pa im se može manjati vrednost u toku programa. Ova osobine je veoma korisne pogotovo što se može primeniti na aveki tip promanijive osim FILE Moguće je tako i inicijalizovati

Ako pravi programeri ne govore paskal, to samo znači da nikada nisu sefi za IBM PC i učitelj u svoj računar „Turbo paskal". Prednosti Borlandovog „Turbo paskala" najbolje dolaze do izražaja kada se uporedi sa ostalim kompajlerima za PC računare. Poređenje ipak nije nimalo lako, jer „Turbo" zastupa sasvim drugačiju filozofiju: pomiriti snagu kompajlera velikih sistema za operativnošću i lakoćom programiranja koja krasi interpretere malih računara.

```
RECORD promenljive, na primer
CONST DEMO ARRAY [1 2] OF RECORD
DEMOI INTEGER,
DEMOS STRING[10],
END=(
(DEMOI 1, DEMOS string 1'),
DEMOI 2, DEMOS string 2')
```

Turbo je bio obogaćen i takozvanim hardverski zavisnim instrukcijama Memorija računara za smetre predefinisanim nizovima MEM (ARRAY OF BYTE) i MEMW (ARRAY OF INTEGER) tako da za može direktno adresirati sa MEM[seg ofs] i ulaz/izlaz sa tretiraju na isti način nizom PORT[0 $FFFF] Postoji i nekoliko procedura (MOVE, FILLCHAR itd) koje omogućavaju direktno obraćanje memoriji što sve čini Turbo paskal' pogodnim čak i za razvijanje sistemskih programa

Standardni elementi paskala su primenjeni gotovo u potpunosti, uz neka finije proširenja kao što je recimo CASE OF pa na kraju ilate i ELSE itd

Tipovi numeričkih podataka su BYTE (0 255), INTEGER (−32768 32767) i REAL (1E−38 1E+38) Ovaj poslednji je predstavljen za 40 bitova mantiza, što daje mogućnost za tačnost na 11 značajnih cifara Fond aritmetičkih funkcija je minimalan ARCTAN, COS EXP, LN, SIN, SOR i SORT Što za tačnosti tiče, proveravali smo detaljno zamo EXP i ARCTAN funkcije koje su pokazivale grešku u poslednjem bitu mantise i to samo oko sredine testiranog punog intervala

Ukupan memorijski prostor za promenljive je ograničan na već uobičajenih 64K Za stek za može odvojiti daljih 64K, dok je prostor za dinamičke varijable ograničen isključivo raspoloživom memorijom

Brza brzina

„Turbo paskal ' uglavnom prevodi program bez posebnih optimizacija Rezultujući kod je dosta kompaktan a obzirom da se, zbog ograničenog prostora, češća oslanja na pozivanje elementarnih rutina u svojoj biblioteci Time sa donekle gubi na brzini, ali su poređenja pokazala da u praksi gubitaka gotovo da nema Na primer, ista petlja sa operacijama nad realnim promenljivama je sa fortran kompajlerom (MS 3 10) oko 10% brža u odnosu na Turbo paskal, ali traba uzeti u obzir njegov za jedan bajt širi format mantise

U ovom trenutku, mogu sa naći dve verzije kompajlera 2 1 i 3 0 Razlike su više interne — u novijoj verziji je red za detotekama baziran na DOS funkcijama većeg broja koje su jednostavnije za upotrebu e nelaze za u verzijama DOS 2 00 i kasniji Uvedeno je i nekoliko manje značajnih novih procedura, ali je novije verzije daleko bolje opremljene Mogu za nabaviti još dve verzije istog kompajlera, od kojih TURBO-87 računanje realnim brojevima poverava matematičkom koprocesoru (uz opseg brojeve do E307) koji to čini veoma velikom brzinom, tipično oko 20 puta brže bez obzira na povećani format broja TURBO-BCD je poslovna verzija kod koje su realni brojevi predstavljeni u desetobzjtnom BCD formatu, što donekle usporava red ali oslobada potreba da za vodi računa o zaokruživanju, jer greške pri osnovnim matematičkim operacijama na postoje Ujedno je ugrađena i naredba FORM koje zamenjuje WRITE i omogućava sve standardne formata lepiza numerika uobičajane u poslovnim primenama

Uz sve to, Borland prodaje i nekoliko disketa za izvornim programima iz pojedinih oblzati Razvijen je kompletan tekst editor niz rutina za rad za grafikom visoka rezolucije (koje podržavaju i Herkulsa grafički adapterj, rutine za kreiranje sopstvenih programa za rad sa bazama podataka, možete nabaviti i izvorni program za šah i neke druge igre, sve u svemu priličan fond gotovih rutine iz kojih može dosta da za nauči.

Turbo paskal u ovom trenutku predstavlja najbolji opštenamenski kompajler za IBM PC računar Mnogim izmenama sa dosta odmakao od standardne verzije jezika i prarestao u posebnu kategoriju koja, sudeći po reakcijama, stiče veliku popularnost Jeden od razloga je svakako i izvanredna prilagođenost računaru koma je namenjen Cena u inostranstvu je oko 70 USA$ za osnovnu verziju i $125 za sve tri varijante

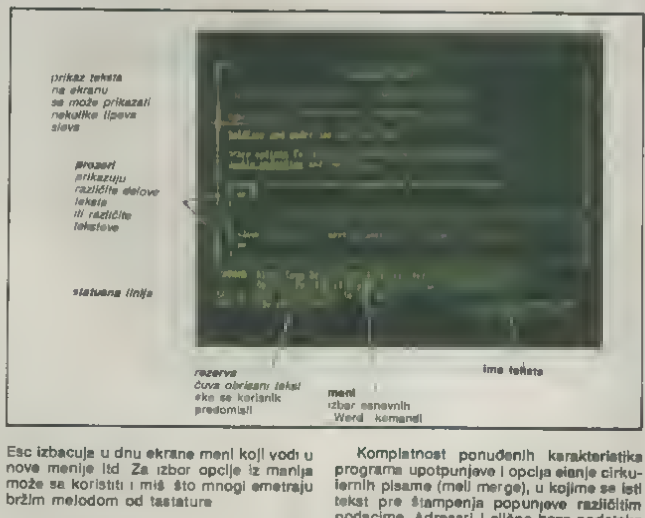
Zoran Životić

treća sreća

Microsoft Word 3

Malo ko na bilo koji način vezan za računare nije čuo za softvereku kuću koja se već godinama nalazi u samom vrhu. Microsoftu dugujamo bejzik mnogih kućnih računara operativni sistem za PC-e i još mnogo drugih vaoma kvalitetnih programa. I pored svega, programom za obradu tekste „Word", nikada nije postigao ioia značajniji uspeh. Iako je program bio veoma ambiciozno zamišljen, nija pružao eve ono što je obećavao, mnoge opcija niau ni bila izvedena, a uz ave to je i rad se njima bio vaoma kompikovan WORD 3, nova verzija atare ideje, trebe ave to da promani.

Program redi ne svakom IBM PC kompetibilnom računeru se minimum 256 K mamonje Pun kontor je moguće poetići samo ee dve disk jedinice Program se prodeje bez zaštite od kopiranje, tako de se ieko može instaliseti ne tvrdom disku čime otpede potraba ze menjanjem diskata u toku rede installranje programe u zavienoeti od respoiožívog herdvara je doste jednostavno zahvaljujući jasnom „SETUP programu Nedoumice se (kako je već uobičajeno) mogu jeviti samo kod izbore vidao edeptara Word se najboije ponaša keda redi se grefičkom kartom jer je predvideno de se već u toku pisenje ne ekranu sagledeve konačni izgled strane uključujući porad formate I tip slova koje se koriete Teko će izbor ne primar Italike ne ekrenu zaista pokazati kurzivna slove Ipek za duge tekstova ova osobine nija od važnosti i nejčeše se koristi običen takst mod Gratičke karte u njemu deju veoma lošu eliku pa je možde jedine prava etternative izbor EGA interfajsa, koji sjedinjuje najbolja osobina oba načina rade a uz to omogućave i povećani formet od 43 rede sa po 80 znekove



Ne pola pute

Word 3 se donakle razlikuje od ostalih tekst procesore I potrabno je odredeno vreme da se naviknete ne njegov „način razmišljanje Formetiranje se obavlje u toku pisanje, pe je ono što se vidi ne ekrenu istovremeno I konačni izgled etrene kod štempenje Ipak ovo nije do kraje izvedeno Na ekrenu se ne pokazuju različita veličina elove kao ni proporcionalno razmicanje To ne smeta kod etandardnih matričnih ili štampeća se ispezom, ali može da predstevlje problem kod laserskih štampača ze koje Word pokušava de se namatne kao nejboiji Ova nova generacije pruža velika mogućnosti, prektično oslobeđa menja izdeveća skuplih foto-slog sistema i stiče eve veću popularnost PC računeri za sade nieu opramljeni dobrim programime (kao što je to ne primer Apple Mecintosh) koji bi podržavali ove izlazne uređaje, I Word je u ovoj oblesti leko delako od savršenog tranutno bez konkurencije Ovakvu orijantaciju Microsoft-a dokazuje I verzije ietog programe namanjene ze red u mraži keko bi više PC-e moglo de dela lati skupi laserski štampeč

Ostvarivenja funkcije programe se odvije pritiskom ne funkcijske tastera kombinecijom slove sa ALT testarom ili kursorskim tasterime na numeričkom dalu tsteteture Za raredbe koja su rede u upotrabi, pritisak ne Esc izbacuja u dnu ekrane meni koji vodi u nove menije itd Za izbor opcije iz manija može se koristiti i miš što mnogi emetraju bržim melodom od tastature

S prozora na prozor

Ekran se deli na prozora tekste se različitim dokumentime Kopiranje se odvija prathodnim prebecivenjem dala tekste u posebno odvojen proetor e zatim njegovim kopiranjem u leti ili bilo koji drugi tekst Fusnote se pišu takođe u posaben proetor koji može biti u nakom od prozora ne ekranu teko de se ime latovremeni uvid I u nepomenu I u tekst ne koji se ona odnoei

Word vodi račune o njihovom numerisanju Može ih štampeti bilo na odgoverajućoj streni bilo na kraju tekste kao ekup nepomene Još jedna rede eretane opcije je I indeksiranje pogievije III delove tekste koji se zetim u obliku sadržaje, izdvejeju u posebnu tablicu

Formiranje lapise se odvije na način koji je uobičajen u ostalim tekst procesorime , Word prati nejnovije trandove pe poseduje ne primer, I višekolonski lapis Veoma korisne osobine je mogućnost da se način formetirenja sačuve na dleku kao podatek o etilu lapise i primeni ne bilo koji tekst, čime se obezbeđuja uniformni izgled dokumeneta bez obzira na to ko ga pliše Krairanja višekolonskih tabilca je veome jednostevno a Word lako uključuje podetke iz bilo kod drugog programe tipično baza podateka kao što je dBase

Kompletnost ponuđenih karakteristika programa upotpunjave i opcija elanje cirkuliernih pisame (meli merge), u kojime se isti tekst pre štampenja popunjeve različitim podacime Adresari i slične baze podataka mogu se kreireti kao tekstovi u „Word-u" I podaci selektivno uključivati u gievni dokument

Provere isprevnosti epelovanje je izvedeno ne poznat način Novost je što „Word-ov „Spelling checker omogućava de se rečnik proširuje bilo generalno bilo da se kreire specifičan rečnik za odredeni tip tekstove Ovim poslednjim se značajno ubrzave red jer sa provera vrši samo za specifične rači

U programi je ugradena I jednostavne matametike koje se može primeniti nad kolona I redove podetaka u tekstu, pa nama potrabe de nepuštete program zbog tako jednostavnih stveri

Ukupan utisak o programu je veome povoljen Program nudi gotovo eve opcije koje vem mogu ikad zatrebati i orijentisan je, pra svege, na profesionalne korienike kao što su pisci I prevodioci, sekraterios neuče institucije i menji izdaveći Njegove vrednoet dolazi do punog izražaje kada se poveže ea laserskim štempečem U ovom tranutku ima oštru konkurenciju, pra evega u programima Mutimete Wordperfect , Displeywrite koji eu već etekli čvrste pozicije I nude slična mogućnosti Cena progreme u Engleskoj ja oko 430 funti

z Ž

Računari iz mog ugla

Kako se oženiti . . . i živeti srećno stotinugodina

Kompjuterska novina prepune su raznih uporednih prikaza računara i vi treba da se opredelite na osnovu hardvara performansi softvera cene Sva to bi svekako trebalo da ima neki značaj ali nikako na lako direktan način Ne smete dozvoliti da računar ne osnovu skrivene reklame ili stvorena psihoze izabere vas a ne vi njega Pri izboru računara morate mnogo više da vodite računa o osobinama vaše ličnosti nego o raznim tehnikalijama i praktikalijama

Najvažniji kriterijum pri izboru računara je da vas učini SREČNIM Nije reč o računaru na poslu Računar na poslu je kao kolega sa posla možete da ga mrzite ali morala se njim da provodite radno vreme A gde je ljubav?

Vi stupate u određeni intimni odnos sa računarom koji je u većoj ili manjoj meri ljubavni lako je računar muškog roda nema apsolutno nikakvih razloga za grižu savesti da radite nešto nenormalno

Moguća bi bila tipologizacija vlasnika na osnovu toga kakav računar neko ima kod kuće ali to se na bi poklapalo se time šta bi trebalo da ima Važno je umeti prepoznati svoju sreću kako kaže Čika Jova Zmaj i životna saputnica sa bire teško Ako ne znate kakva vam žena trebe moći ćete da izaberete ne osnovu svog računara i obrnuto ili ćete bar znati kakva vam žena (računar) ne treba

Čak i da ste izvršili idealni izbor na oba plana ne znači da će vam utroje biti lepo Računar se očito maša u ljubavni život Do sada sa kao razlog za razvod braka nije pojavljivalo kompjutersko zanemarivanje (okrutnost) Problematiku kako harmonično živeti sa ženom i računarom ovog puta ne razmatramo Ograničićamo se na izučavanje izolovanog uticaje dva tela (vas i računera) jer kao i u nebeskoj mehanici problem tri tele u opštem slučaju nije rešen

Izbor sa obično vrši na dve načina

Prvi je poznat pod nazivom sliku svoju ljubim a pogodan je i donekle u redu za one koji su dobri lepi i pametni — ako ste ružni ručno ćete i odabrati Za razliku od partenogeneze bolji način predstavlja izbor komplementa. Tako ste na dobitku Vi treba da uskladite sebe i računar u harmoničnu zajednicu da se uzajamno prožimate i dopunjavate

Sve se može birati po ovim principima ne samo šeputnica nego i kućni ljubimac — od papagajа do računara

Svaki izbor je kompromis između realnih stvari i onog što vem duša ište Nevolja je u tome što se pravi previše kompromisa i svi kupuju iste stvari

Ime računara koji su na glasu kao dobrn prilika mada koriste mnogo šminke Na vodno mogu sve Tu je nešto sumnjivo Za sada vas neća biti sramota da se pokažete sa njim u društvu ali nesamo ćete samo vi znati kakav je Kada vam bude u rukama bićete u stanju da ocenite koliko stvarno vredi Može vam se dogoditi i da vas razočara jer ste mnogo više očekivali Ne nasedajte na jeftine intelektualne tipove sa brzih večernjih kurseva koji se dobro prilagođavaju vama To je najbolji primer kako vas kupuju Ako je amstrad računer vašeg života čuvajte se devojaka uz koje kao paket-aranžman obično ide i cela njena familija

Može vam se desiti i da nasadnete na drugu krajnost i nabavite računar koji svi imaju koji je zato izgubio svaku izuzetnost Mašinu kojoj je fascinantnost narušena moraćeta pod hitno da menjate ili zato što su je mnogi imali ili zato što su i drugi uvideli de ona nije bog zna šta Recimo da ste snob U tom slučaju ne smete imati računar koji se prelerano mešao sa običnim svetom Svake šuša ima komodor a tek spektrum !

Spektrum nikako nije dobar izbor a oni koji se odlučuju za kupovinu u današnjim uslovima spadaju najverovatnije u kategoriju račundžije koji su nešto loše izračunali Ako vam ja bar malo stalo do socijalnog presliža, biće vas sramota da siroti spektrum iznesete pred goste Potrudite se da pri izboru životna sapunice ne počinite istu grešku jer dok je sa spektrumom mogući će ljubavna avantura gde ćete se malo polgrati njime i ostaviti ga čim vam se ukaže bolja prilika sa živom ženom je to daleko teže Tada imate dve mogućnosti

da je se nekako otarasite, ili da se posluzi tehnikom maskiranja kao što rade mnogi spektrumovci Dok oni kamuflir spektrum u razne veće kutije sa profesionalnom tastaturom vi možete svoju zen da kupite bundu

Masovni izbor uopšte nije izbor za odbijam da o toma govorim

Totalna je greška kupovati i računa nulta socijalna kompanzacije Smatra se da je to sirotinjski izbor Mnogo greše školske vlasti kad nabavljaju deci iz pasivnih kra va inferiorne mašine ukoliko ne žele da obeleže za ceo život Da ne pominjem roditelje koji su ekstremne stipse pa kupuju ovaj računar deci jer je šteta da pokvar neki skupiji Od takve dece posteju ljudi sa prilajenim osećajem inferiornosti Traumatična iskustva sa ovom vrstom računa vuku se kroz ceo život Osim toga m vas deklarisati kao mazohistu koji doživ va sladostrašće sveki put kade pokuša da ukuce neki duži program preko nacrtane testatura

Za vas nisu ni promašeni modeli To je izbor za mala pare Vi ionako ne mislite da učite programiranja Nečete valjda da vas upoređuju sa takvima koji se razmeću da kod kuća imaju računera komade 5 (pet ni jeden ne umeju da upotrebe

U ovu grupu ne spadaju promiskuitetni tipovi koji vola da probaju što više računara Ležerno ča da se hvale da su celu noć radili na emigi Takvi što daju prednost kvantitetu, u ljubavnom životu teže ostvare što više racki

Ako prednost dajete kvalitetu i žudite za tim da budate fascinirani izbegavajte računare koji mnogo obećavaju Vaša radost biće kratkog veka, jer takvi računari većini slučajeva ne ispune obećanja. Moraćete da ih menjata pre nego što drugi primeta da ste se zaznuli

IBM PC po definiciji ne dolazi u obzir mada već sam po sebi predstavlje dokaz da u računarskom smislu niste mekar ko mašina koja vas neće osramotiti ali uskoro biste ustanovili da nije baš mnogo zabavna. Uostalom za ono za šta biste je vi koristi može da posluži i neka mnogo jeft nija igrečka

Očigledno ze vas nisu ni masovni jeftini ni ekskluzivni modeli Vreme je da vam otvoreno konkretno i bez uvijanja predočimo pravi računar za vas

Jelene Rupnik

(NASTAVIĆE SE)

procesor naših snova

RISC

Od trenutka kad je prvi centralni procesor ugledao svet (odnosno ieboretoriju), svi su se proizvođači trudili da netrpaju što više neredbi u svoj čip bez mnogo razmišljanja o tome da li je korisnicima sva to potrebno ili ne. Tako su vremenom procesori (a potom i mikroprocesori) počeli da množe, dele, pozivaju procedure, stvaraju lokalne verijable i rade slične komplikovane poslova koji su, u stvari, nasleđeni iz filosofija današnjih viših programskih jazika pa shodno tome, ne vode mnogo računa o efikasnosti. Zauzvrat, daju mogućnost da i dosta loš programar piše prevodioce za više jezike (pa se posla korisnik čudi kako to da njegov C daje tako dugačak kod ili tako katastrofalne numeričke rezultate). Dugo vremena je tehnološki razvoj čipove bio u velikoj ekspanziji pa drastično povećanja gustine pakovanje ili brzine izvršavanja nije predstavljalo neki problem. Čak su se firma ne sev glas hvalile čipom sa recimo 100 000 trenzistore, smatrajući de je to samo po sebi dokaz vrhunskog kvaliteta. Takve je filosofija našla naročito ohrabrenje u gradanju ilap-procesore koji „direktno izvršavaju izuzetno visok programski jezik. Tržište je rselo: troškovi padeii i sve je bilo ružičasto.

a onda je puklo

pred očima nekih programere (u prvom redu ekademekih) koji su shvetili de procesor koji ime neredbe-monstrume, čije izvršenje zne de potraja preko 150 vremenekih ciklusa, u vrema kad granice gustine pekovanje i povećanja brzine postaju vrlo uočljiva i čvrste, na može još dugo de zadovoljava sva veće potrebe u redu. Posla takvog „otrežnjenja" počeo je vrlo intenzivan rad na minimiziranju broje i složenosti naredbi ili preciznija rečeno emenjivanju broja kodova koje procesor respoznaja. Na prvi pogied bi se reklo de neme svrhe previti razliku izmedu broje kodova i naredbi. Medutim, eko znemo de MC 68000 ima tek nešto više od 55 neredbi e raspoznje preko 1000 kodove (zbog vrlo velikog broja adresnih modove) postaje sesvim očita rezika ove dva pojme.

Prvi završen procesor se pojavio pod krovom Hewlett-Peckarda što i nija naročito iznenadujuće s obzirom ne ranome ova firme. Nija se odmah nešao u prodeji zbog potreba razvoja kompletnog računerekog sesteme (HP nikad nikome nije prodeo samo komponantu već uvek završen elstem), pa ja promocije bile tek u februsru ove godina. Samo den ksenije IBM je dosta bučno najavio svoj modal čima je filosofije računera se smanjenim skupom instrukcije (RISC-Reduced Instruction Set Computer) postala definitivno „legeine" što je izbilo mnoge argumenta iz ruku protivnika ovog koncepte koji je iz ekadamskih prešao u

proizvodne stara. Budući de ja procesor Hewlett-Packarde prvi, dosad najbolje predstavljan, a kako stvari stoje i najbolji (toliko da ge ne nazivaju RISC već HP Precision Architectura) upoznećemo RiSC-filosofiju na njegovom primeru. No pra toge se moramo podsetiti nakih retko pominjanih „sitnica" vezanih za

stare dobre

klasične procesore, koji ipak neće otići tako brzo u staro gvožde. Klesični procesori imaju dva vrste internih instrukcija: cenovne, koje su ugredane u hardvar (obavlje ih ALU-Arithmetic and Logic Unit), i kompleksne koja su potprogrami sastavljeni od osnovnih. Takav procesor, dakle ime unutrašnju strukturu kao bilo koji računar, pa zahteva dosta dodatnog vremena za procesirenje eveka neredbe i raspoznavanje adresnih modova, pa potom za čitanje internog ROM-a. Mašineka neredbe dostupne korisniku su, naime, opet potprogremi internih instrukcije (koje se tradicionalno nazivaju mikrokod). Neki procesori (i vrlo retki mikroprocesori) omogućavaju korisniku pisanje i mikrokodirenih programe, no to zahteva dodatni razvojni elstem i mnogo ekstre-brze (i ekstre-skupe) memorija, dok gubitak vremene za dekodiranje i dalja ostaje. Postoje procesori koji se greda u tahnici brt-odrazaka kod kojih korienik od početka stvara sva mašineke neredbe (pa, recimo, eveki kompejler raspolaže „svojim" mašinskim neredbeme), no ta je tehnika izuzetno skupa i procesori tog etila neće nikade dobiti prefike mikro- tolike potreban za avantuelnu dominociju. Osim toge, organizacija procesora se mikro i mašinskim kodom nosi u sebi nepotrebno opterećenje, jer se, praktično dva puta radi leti posao ne dva različite jezika, e tim da je mikrokod nedostupan recimo C-prevodiocu, e mašinski jezik nedostupen elstemu za razvoj mikrokoda — dakle nešto kao gluvi telefoni ili medurepubličko dogovarenja. Stoga RISC filosofija zahteva de instrukcije procesore budu

sve u hardveru

dakle bez ikakvih mikrokodova, internih ROM-ova i sličnih raspinika vremena. Instrukcije su, nerevno, jednostevne (one koje se najčešće i koriste), pe im ne treba nikakva interne poštapalica de bi se izvršavale. Drugi važan imperativ je da eveka instrukcije „potroši" samo jedan ciklus sata. Od ovog su izuzetne instrukcija za skok (Brench) uzimenja (Load) i ekiedištenje (Store) podataka. Medutim, eveka naredbe koriet stvarno tri ciklusa (tatch, exectue, store) ali se oni preklepeju (vidi sl. 1) tj. koriета se tekuća-iinije (pipeline) koje omogućava istovremeno izvršavanja tri faze (cikluse) tako de je ukupni efekat kao da je sveka instrukcije „potrošile" samo jedan ciklus. Ove tehnika je odevno pozneta u svatu procesore (pa i nekih mikroprocesore) i zahteve takozvanu „cashe" (keš) memoriju ultre brzu unutrašnju mamoriju koje čuve instrukcije i podatke koji će kasnije biti obradiveni i tako emenjuje potrebu za komunikacijeme se sporim epoljašnjim čipovime. Kompletnu strukturu High-Precision Architecture procesore vidite ne elici 2. Boiji poznevaoci arhitekture procesore će odmeh primetiti nedostatak mikroprogremekog siedine i mikroprogremska memorije, kao i činjanicu de koprocesor pokretnog zareza ima direktan pristup u cashe koji se sastoji od odvojenih delova za podatke i neredbe što nije nepozneto eli ni često viđano.

Zahvejujući svim tim hardverekim „sitnicame" procesor je izuzetno brz pa se kompleksna instrukcije koje se kod klasičnog procesore nelaze u mikrokodu mogu pisati u

milkodu

ovde se ne radi samo o igri reči (odnosno prefiksa), već o suštinskoj razlici. Milkod je, naime, softvereka tvorevina, koju procesor prepoznaje kao običan potprogram, potpuno revnoprevan svim ostalim potprogramima i sačinjen od istih mašinskih naredbi kao i eveki drugi. Takav milkod se može pisati u istom višam programekom jeziku kao i ostali - obični jezici, e jedino ograničenje je de sma da menja samo ona procesoreke registre koje uzime kao ulazne parametre i nake opšte namene, mada je i ono na nivou korienika, to jest procesor ne vrši nikakvu provaru u toj sferi. Dalje prednost milikode je što se dodatno vreme troši samo kace se neki takav potprogram pozova (dok se kod mikrokodirenog procesore dodatno vreme stalno troši ne odložano dekodiranje). Budući de je softvereka tvorevine milikod je prenosiv ne mnoga mašine iste klase (što će naročito u budućnosti biti značajno) a različitih procesore, što kod mikrokodirenih mašina nije

Još od pradavnih vremena (u računarstvu vremena brzo postaju pradavna) u kojima je Alan Turing sanjao o računaru beskonačne brzine i memorije, pa sve do danas, svako ko se ikle više bavio računarima nosio je u sebi neki svoj san o idealnoj mašini. Pojavom svake nove generacije računara, nekima su snovi bili lepunjeni dok su drugi nastavljali da sanjaju do nekih novih generacija. U poslednjih godinu dana pojavili su se procesori koji zaista izgledaju kao iz nečijeg sveta mašte.

bilo ostvarljivo ni u najludim snovima. Neki snovi, dakle, ipak postaju stvarnost.

Tekuća linija, koja je jedan od stožera postojanja RISC-filosofija uopšte, postaje potpuno neefikasna kad program sadrži mnogo skokova. Jer se gubi jedan ciklus na dobavljanje instrukcije sa neke dotada nepoznate adrese, tako da, racimo obične petlje sa per hiljada ponavljanja (recimo, pri pretraživanju baze podataka ili takata) može potpuno da razori tekuću liniju. Ovaj procesor koristi jedan duhovit trik o koma će kasnija biti reči kad budemo razmatrali softverske karakteristike a sada pogledajmo

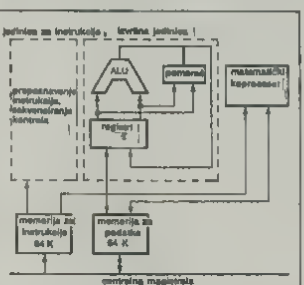
programski model

kako to ovajani softveraši vole da kažu. Na raspolaganju su 32 registra opšte namene. 32 kontrolna registra (od kojih jadan statusni) i 8 prostornih registara. Registri opšta namene nemaju sa sada nikakvu specifičnu ili praporučenu namenu. Dužine im je 32 bita i prevodilački sistem ih koristi za prenos parametara izmadu procedura. Kontrolni registri se koriata za procesiranje prekida, zaštitu virtualne memorija i druga sistemska funkcije. Stanja procesore se odražava u 32-bitnom statusa registaru. Prostorni registri specificiraju 4 gigabajte dugačke virtualne prostore i sadrža 16 ili 32 bita, što daja ukupno 48 ili 64 bita virtualnog adresnog prostora. Dva adresna registra takode pokrivaju maksimalno 64 bita virtualnog prostora, što je u sadašnjim uslovima praktično neiskoristivo, ali garantuje da će ovaj procesor još dugo zadovoljavati sve potrabe korisnika. Virtuelne adrese se prevode u fizička posredatvom specijalnih odvajača takozvanih TLB (Translation Lookaside Buffer), koji su takode hardverski rešeni. HP Precision Architecture podržava rad sa više tipova podstaka oslobrojni su dužine 16 bita (halfword), koji moraju biti postavljeni na parne adrese, 32 bita (word) koji moraju biti postavljeni na adrese deljive se 4. Karakteri su dužine 8 bita i zadovoljavaju ASCII standard za vrednosti %-127 i HP-ov proširani Roman 8 set za vrednosti 128-255.

Decimalni podaci mogu biti pakovani i nepakovani, ali su uvek postavljeni na granicu reči (početna adrese deljiva se 4) i mogu se sastojati od 7, 15, 23 ili 31 BCD cifra.

Brojevi u pokretnom sarezu se predstavljaju u skladu se ANSI/IEEE standardom. Podržana je jednostruka, dvostruka i četvorostruka preciznost (vidi sl. 3), odnosno obezbeđena 23, 52, 112-bitna (7, 16, 34 dekadnih cifara) tačnost. Operacije u pokretnom zarezu se mogu softveraki simulirati ili izvoditi na posabnom FP procesoru koji (budući da direktno postupa u „cache") na oduzima praktično ni vrema ni

Instrukcije se pribavljaju i izvršavaju u tri ciklusa, ali se oni preklapaju u tzv. tekućoj liniji pa je efektivno vreme trostruko kraće



Procesorski modul

dodatnu memoriju. FP procesor ima dvanaset 64-bitnih registara sa podatke, 32-bitni status registar kao i sedam 32-bitnih registara za procesiranja grešaka-izuzetaka (tzv. IEEE exceptions). Raspoznaja 11 naredbi (FADD, FSUB, FMPY, FDIV, FREM, FSQRT, FRND, FCMP, FABS, COPY, CONVERSION) u tri preciznosti.

Pravom programeru (a čitalac „Računara" je kao što znamo, pravi programer) se upoznavanja procesora nedostaje još samo

set instrukcija

koji se sastoji od 140 različitih kodova (vidi sl. 4) fikanog formata i dužina jedne reči. Do pre dve godine ovakav procesor bi mogao biti nazvan „velikim raspinikom", ali se danas, kada čak i kućni računari rada na 16/32 procesorima sa 1 Mb jeftinog RAM-a može reći samo „to je ono pravo i posle ledano hladne COCA-COLE" pogledati šta se sva nalazi u tih 140 kodova.

Za rad sa memorijom koriate se samo Load i Save naradba, koja uzimaju 24 koda i povratnano, više od jednog ciklusa, mada dobar pravodilac i tu može dosta da uradi pamatnom raspodalom posla.

Uslovni skokovi pokrivaju 16 kodova. Taj se skok često može eliminisati budući da mnoga instrukcija omogućavaju uslovno izvršavanja (u zavisnosti od nakog bita stanja) u samo jednom ciklusu. Da se pri skoku na bi gubio jadan ciklus i tima rušila funkcija tekuće linije- procesor sa vrema preračunavanja adresa izvršave prvu sledeće naredbu, pa je samo stvar prevodioca da prvu naredbu TEHN-konstrukcije ili povratna petlja postavi neposredno isa naredba skoka. Jednostavno, ali se trabalo satti.

Aritmatičko-logičke naredbe su svedene na najnužnije, ali ipak sauzimaju čak 53 koda. Tu su, naravno, ADD i SUB naredbe kao i naredbe za uporedivanja, te OR, AND, XOR, NOT, poznate sa ranijih procesora. Novost je naredba sa pomeranja i sabiranja istovramano (i sve sa jedan ciklus) koje drsatično ubrzava množanja, naročito malim oslobrojnim konstantama, te EXTRACT /DEPOSIT naredbe za afikasnu podršku bit-manipulacijama, koja koriste viši programski jazici.

Instrukcije sa kontrolu sistema koriste 31 kod i vrše manipulaciju kontrolnim registrima „cache -om, TLB-ima i prekidima. Takode, vrše nadzor pristupa virtualnoj memoriji i rada ostala kućna poslova.

Jedna od bitnih razlika izmedu ovog i drugih RICS procesora je grupa instrukcija za podršku koprocesorima (9 kodova) mogućnost direktnog pristaka na rad u pokratnom zarezu kad i četiri „specijalne operacija koja rada ono što korisnik (meda će to češće biti Hawlatt-Packard) zaželi dakle nešto kao čitač naših anova.

Kao što se vidi nija zaboravljena nijedna važna instrukcija ali nija bilo ni razbacivanja. Dolazimo do logičkog pitanja ko je taj sveznajući monstrum koji može sa sigurnošću da tvrdi da su baš tih 140 instrukcija najpotrabnije i da se najčešće koriata? Pa, naravno to je

spectrum,

ovda se, svakako, na redi o Sinklerovoj „maloj-ružnoj-tupavoj" igrački, već o ogromnom razvojnom projaktu u koji je Hewlett-Packard uložio 100 miliona dolara što je i za američka prilike velika suma. U okviru tog projekta je izvršena analiza mnogo megabajta mašinskog koda iz raznih aplikativnih programa da bi se dobila frakvencija korišćenja pojedina vrste instrukcije. Potom su, uz konsultaciju hardverača, među njima izabrane ona instrukcija koja se mogu izvršavati u jednom ciklusu. Tome su još dodana neophodne instrukcije akoka i referanciranja memorija (gda je opet presudnu ulogu odigrala učestalost korišćenja). Glavne snage ovog projekta je što su softverski deo (tj. kompajlerski sistem) hardverska baza razvijani istovramano. Koristeći simulaciju još nepostojećeg procesora, programeri su mogli još u ranim fazama razvoja da koriguju naka od planiranih instrukcija. Osim toga, procesor je po završetku razvoja u startu raspolagao sa 4 programska jazika izvranim optimizatorom sa tri nivoa i moćnim MPE XL operativnim sistemom. Čitava tahnologija je najvaljena kao arhitakture za 90-ta i ksanija podina što znači da firma neće tek tako ispucati ave aduta, tj. da program Spectrum ostaje dalja aktivan. U stvari, ono što se do sada vidalo je samo simulacije procesora serije

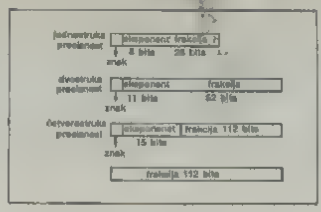
3000 čuma je dokazano (što je vrlo bitno za buduće poslovne odnose, a što su neki osporavali) da HP Precision Architecture može uspešno da podržava poslovne aplikacije i kobol. Ček i u svom „biedom" izdanju se procesorom u TTL diskretnoj tehnologiji performansa su izvrsne.

Prosečna brzina od 4 5 i maksimalna od 6 MIPS-a rečito govore da će izgleda i moćni IBM ovog puta položiti oružja pogotovu kad na scenu izađe VLSI verzija procesora kad glavne aplikacije budu na C-u i kompajleru i kad u prodaji bude novi prevodilački sistem koji psakal fortran kobol i C prvo prevodi na spektrumov međukod SPELIC (Spectrum Low-Level Intermediate Code) tako da se optimizacija i konačno oblikovanje izvršnog koda (koji se može optimizirati na još dva nivoa) izvodi jedinstvenim procesom nezavisnim od jezika koji se prevodi. Taj je prevodilački sistem u stvari baze na kojoj počiva prava efikasnost RISC-tiosotija budući da softver brine mnoge „hardverske" brige pa će čitaoci Računara imati priliku da se posebno sa njim upoznaju a sad je red da spomenamo

ostale

predstavnike „new wave"-a. Prvi je (po datumu pojavljivanja) Transputer T414 koji je na granici RISC-a i procesore uopšte budući da predstavlja mikrokompjutar-na-čipu. Sadrži (porad CPU-a) i 2K RAM-a dva DMA kontrolera i četiri serijska I/O kanala sa brzinom prenosa od 10 Mboud-a. Sam CPU ima samo 6 32-bitnih registara od kojih su samo 3 slobodna za rad. Međutim pošto je RAM u čipu brz kao i sam procesor na raspolaganju je u stvari 500 32-bitnih registara. Brzine jednog čipsa je 5—10 MIPS-a. Posebna pogodnost Transputera je lako i efikasno povezivanje u mrežu dakle i ova tehnologija cilja pre svega u budućnost koja je dosta ružičasta (ako je inmos malo (čitaj finansijski slabe) firma budući da već postoje razvojni paketi za VAX i IBM PC te da je u prodaji i inmosov razvojni sistem kao i specijalni jezik Occam koji maksimalno podržava multiprogramski i multiprocesorski rad i u kome su napisani i paskal fortran i C-kompajleri. U prodaji su takođe i tri razvojne pločice od kojih je jedna IBM PC-kartice što je svakako veliki mamac za buduće korisnike, a samim tim i velika početna prednost ovog procesora u odnosu na sve ostale.

Ako se neko do sada još i pitao zašto je Acorn pao u velika finansijske teškoće, njihov ARM (Acorn RISC Machine) sasvim pouzdano daje odgovor. Čip je razvijen za samo 18 meseci (Hewlett-Packerd je svoj razvijao 5 godina a IBM nešto duže ali se prekidom) a urađen je u zastarelo tromi-krometarskoj tehnologiji. Pored toga što ima samo 25000 tranzistora i svega 44 instrukcije postiže brzinu koja je svega 0 5 MIPS-a veća od brzina MC 68020 (koji predstavlja kompletnu mašinu kompleksnih mikrokodiranih instrukcija) pa se čak i slabije upućenom posmatraču namače pitanje zašto se uopšte ušlo u gradnju čipa koji je već u startu zastareo, a uz to bez



Formati računanja u pokretnom zarezu

ikakve softverske podrške se jednom šansom da postane spoljni procesor BBC a (naravno ako se nađe neko dovoljno lud da za to da 2000 funti).

IM ja očigledno, bio zatečen brzim izlaskom sklepanog ali uprkos tome brzog HP računara teško da je brže-bolje morao da i sam izbaci svoj računar IBM-RT PC koji vrlo solidno izgleda (a toliko i košta). O njemu ja poznato vrlo malo podataka ali je najbitnije da raspolaže procesorom sa 118 instrukcija pri čemu su 84 jednociklusne koji pri svega 5 88 MHz dostiže brzinu MC 68020 na 12 5 MHz, što rečito govori o velikoj brzini (naročito kad izađu modeli sa većim klokom). Računar ima IBM-ovu verziju UNIX-a što znači i obilje aplikativnog softvera kao i upravljač memorije koji adresira da 1 Tb virtuelnog prostora. Za sada je namenjen prs svega za CAD. Osnovni model (CPU 1 MB RAM-a 40 Mb tvrdi disk i monohromatski monitom) košta oko 14 a najjači (2 Mb RAM-a, 70-Mb tvrdi disk i kolor monitor) oko 40 hiljada dolara.

Poslednji procesor izgleda, ne prvi po gled kao uljez među RISC mašinama. Radi se o procesoru koji izvršava visok programski jezik Fort ali to čini sa svega četrdeset tak naredbi postižući maksimalnu brzinu od 10 MIPS-a a minimalnu od 6 (za diskretnu verziju procesora) i sadrži 2 K internog RAM-a. U čemu je stvar? Fort je relativno mlad jezik koji je tek pre neku godinu pokazao da može da bude ravnopravan se lispom u razvoju veštačke inteligencija a moćniji u svim drugim primanema naročito u real-time aplikacijama. Mada je potpuno strukturiran i visoko modularan on je brz skoro koliko i mašinski kod (sada on i jeste mašinski kod) i što je najbitnija vrlo lako vaskrsnave iz svog pepela tj. dovoljno je četrdesetak instrukcija da se postavi ceo jezik (koji sam sebe proširuje) pa je proćto idealan za primenu RISC filosofije. Ovaj procesor uopšte ne mora da brine za razvojni softvar i operativni sistem budući da je fort sam sebi operativni sistem, a softvera ima u izobilju i po vrlo povoljnim cenama. Kad izađe VLSI varzije sa izmanjenim (čitaj višestruko ubrzanim) sistamom dokodiranja mogu sa očekivati i brzine bliske 50 MIPS-a, što uz visok programski jezik može da znači i novi veliki proboj u području veštačka inteligencija. Do pojave čipa u prodaji, softvar sa može pisati na bilo kom kućnom računaru (ps ček i na Sinklerovoj igrački) što je još jedan bitsn faktor za priviačenja velikog broja muštarija i naravno pad cene proizvoda. Procesor proizvodi kompanija Metatorth čiji su osnivači Alan Vinfild (Wintiald) i Rod Gudmen (Goodman), nadafako poznati fort-ekspartl (ili siva eminancija ako se to nekoma više svida).

Žarko Barberski

Besplatni modemi

Ukoliko se učlanite u evropsku kompjutersku mrežu Micront i uplatite jednogodišnju članarinu dobićete potpuno besplatan modem *Prism 2000* i prateći komunikacioni softver za vaš računar! Godišnja članarina se sve hardverom i softverom sada iznosi samo 50 funti.

Micronet planira da privuče nove članove još jednom atrakcijom pripremljena je višekorisnička igra *Shades* u kojoj ćete se sukobiti sa 1 000 kompjuterski kontrolisanih igrača i nepoznatim brojem salgrače koji su u tom trenutku priključeni na mrežu. Prvom igraču koji ostvari sve ciljeve igra može da pripadne nagrada od 5 000 funti! (D R)

Dizajnirate li čipove?

Ukoliko se bavite dizajnom integrisanih kola pomoći ćemo vam da potrošite 7 500 funti! Firma *Qudos* za ovu sitnu sumu prodaja trideset dvobitni računar *Cambridge Workstation* sa 20 M hard diskom specijalnim monitorom i kompletnim softverom za projektovanje čipova. Kada dizajnirata integrisano kolo pošaljita disketu *Qudos-u* i za nedelju—dve očekujte prve primerka!

Da ne zaboravimo — kupovinu možete da obavite javljajući se na telefon (0223) 862 333. (D R)

PCST mreža

Ne radi se o šaputanju. Ako za svoju malu poslovnu primanu nikako niste mogli da se odlučite između spora legende sa gomilom programe i jeftinog moćnog klinca sa malo programa, postoji nada za vas. Compuslink iz Kanade je proizveo imaginet mrežu pomoću koja se mogu spajati PC i ST. Osnovni elamanti mreže koštaju samo 900 dolera a šta je to (pogotovu ako tirma plaća) ako ste već morali da platite ST i PCje. (B Đ)

Muvajte grafiku

Ako smatrate da je za sta kriminalno kako vaš dragi računar sporo operiše sa gratičkim elementime na ekranu možete da se razvedrite. Intel je razvio gratički koprocesorski čip 82786 koji može da manipuliše prozorima i sličnim stvarčicema i do sto puta brže nego što se to postiže softverski. Čip sadrži dva procesora i cena mu je za sada nija poznata, iako nije verovetno da će biti mala. (B Đ)

Dejan Ristanović

Dejanove pitalice

Brzopotezni jezik

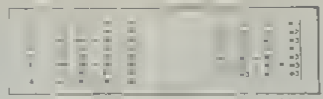
Jedanaesta Dejanova pitalica predstavlja dobru potvrdu istine da novi programski jezik nije lako naučiti čak i kada se on sastoji od svega nekoliko naredbi — u predvidjanom smo roku primili solidnih 235 odgovora, ali je u konačan izbor ušlo svega 155 pisama!

Najpre ćemo se po običaju podsetiti na problem — u Računarima 18 smo objavili interpreter programskog jezika koji smo nazvali *Microalc 86*. Zamišljeni računar ima mamoriju od čitave 3 ćelije čiji je kapacitet za divno čudo, beskonačan u svaku se ćeliju može smestiti proizvoljno veliki ceo broj. Naredbe jezika imaju opšti oblik *Mi=Mj Mk* gde tačke označeva bilo koju od operacija sabiranja oduzimanja množenja i celobrojno deljanje, dok *i, j i k* označavaju mamorijske ćelije ($1<=i,j,k<=3$). Uz zadatak smo predvideli i sliku na kojoj smo ispisali dopuštene operacije *Microalca 86* ali ja ta slika ne nepoznat način izbledela pa je premda malo kasno objavljujemo u ovom Računarima. **Vrlo** je važno da zapazimo da ove operacije oblika *Mi=Mj Mk* **nisu** dopuštene kod oduzimanja i deljanja rezultat se obevazno smešta u ćeliju iz koja je uzet prvi argument. Naredbe *M2=M1/M1* ili *M3=M1/M2* dakle **nisu** dopuštene što je bilo jasno pomenuto kako u zadatku tako i u objavljenom interpreteru. Čak 39 čitalaca nije primetilo ovu začkoljicu pa smo primili otprilike dvoliko rešenja se 6 naredbi ova rešenja amo na žalost moralI de odbacimo jer ne ispunjavaju uslove zadatka!

```
M1 = Mj + Mk,    1 <= i j k <= 3
M1 = Mi   Mj,    1 <= i j   <= 3
M1 = Mj * Mk,    1 <= i j k <= 3
M1 = Mi / Mj,    1 <= i j   <= 3
```

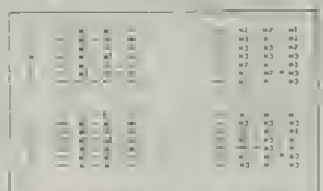
Slika 1

Pošto smo naučili *Microalc 86* počećemo da rešavamo zadatak. U m1 je upisan broj točkova koja neka tabrika ima u magacinu (ova) je broj veći od nule) dok su u M2 i M3 upisana nule. Trebe da napišamo program koji će u M2 upisati broj automobila koje tabrike može da proizvede a u M3 preostali broj točkova — podrazumeva se da automobil ima četiri točka pramda bi bilo zanimljivo razmotriti i realnu situaciju u kojoj kola imaju i rezervnu gumu!



slika 2

slika 3



slika 4

Najpre ćemo zameniti točkove brojevima — naš program trebe da upiša *int (M1/4)* u M2 a zatim i *M1−M3*4* u ćeliju M3. Vidimo da nam je naophodne konstante 4 koju ćemo formirati primenom tri naredbe npr *M3=M1+M1 M3=M3+M3 M3=M3/M1* (pri tome amo koristili zadatu činjenicu da je broj točkova u magacinu veći od nula — inače na bismo dobili broj 4 ćak ni kada bi računar smetrao da je 0/0=1!) Za odredivanja broja automobile nam treba jedna (nedopuštena) naredba *M3=M3/M1* a za tim lako dobijamo i preostali broj točkova *M3=M2*M3 M3=M1−M3*. Tako smo do bili program od 6 naredbi koji koraktno rešava problam ali koji je na žalost napri manjv na *Microcalcu 86* korista se jedno nedozvoljeno daljanja i jedno nadozvoljeno oduzimanja!

Jednostavnom transformacijom rašenja koja smo upravo napisali dotazimo do programe sa slika 2 — dodali smo dve naredbe i omogućili da argumenti za deljanja i oduzimanje budu u M2 i M3 tako da se koriste isključivo dozvoljane naredbe. Na sličnoj se varijanti zaustavilo 35 rešavača.

U postavci smo pomenuli da sa zadatak može rešiti i sa 7 naredbi kako da eliminišemo jednu? Treba zameniti jedno oduzimanja sabiranjem ili jedno deljanja množenjem. Celobrojno daljanje ne možamo da svedamo na množenje ali bi se zato oduzimanje dalo svasti na sabiranja — setimo se da je *A−B=A+(−B)* i da ćelija *Microalca 86* mogu da primaju kako pozitivna tako i negativne brojeve. Umesto konstante 4 ćemo dakle formirati −4 i tako izbeći potrebu da pripremamo M3 za prijam razlika. Ovom malom izmenom dolazimo do konačnog programe sa slike 3 — najkraćim mogućim rešenjam do koga je došlo 155 čitalaca Računara.

Boris Vrabec iz Zagrebe je problam proučio daleko daljanjije od opisa koji smo objavili i dokazao de se zadatak **na može** rešiti sa manje od 7 naredbi. Boris je štaviše dokazeo da postoji tačno 28 koraktnih rešenja sa po 7 naredbi od kojih je sedam aušrijski različito. Rezultate razmatranja koje su *Borisu Vrabecu* donala prvu nagradu od 10 000 dinara vidimo na slici 4.

Drugu nagradu od 5 000 dinara dodelili smo *Ani Kuzmanović* iz Niše koja je poslala po našoj oceni najpopularnije obrazloženo rašenje problema. Treća nagrada od 3 000 dinara je posle improvizovanog izvlačenja pripala *Nenadu Sikimiću* iz Splita.

Trinaesta pitalica

Urednik ove rubrike mora da prizna da nije lako pripremiti po jednu koliko -toliko interesantnu pitalicu mesečno. Trinaesta je pitalica međutim pripremljena čak i pre prve — od samoga sam početka znao koji će zadatak poneti ovaj (ne)srećan broj! Možda će vam sa učiniti da je zadatak previše lak ali — dajmo šansu i mlađim programerima.

Pokušaćemo da pronađemo smisao verovanja da je petak trinaesti baksuzan dan izračunajmo dakle koliko ima ponedeljaka trinaestih utoreka trinaestih sreda trinaestih i tako dalje — videćemo da je petak dan koji je najviše puta trinaesti! Petak 13 je naravno morao da bude proglašen za baksuzan dan da bi se potvrdili Murfijevi zakoni!

Jeste li razmišljali o tome da se kalendar ponavlja svakih 400 godina? Ukoliko dakle ustanovite naku zakonitost kalendara koji važi od 1600 do 1999 godine možete da smatrate da ona važi i za celokupan tok istorije! Napišite dakle program koji prebrojava ponedeljke utorka sreda četvrtka petka subota i nedelja koji su trinaesti u mesecu između 1600 i 1999 godina a zatim upišite rezultata u kupon koji prilažemo ili u njegovu fotokopiju.

Rešenja problema pošaljite na adresu „Računari" (za Dejanove pitalice) Bulevar vojvoda Mišića 17 Beograd tako da pristigna pre 1 decembra 1986. Najboljim rešenjima i najsrećnijim rešavačima će pripasti novčane nagrade od 10 000, 5 000 i 3 000 dinara.

U godinama 1600 1601- 1602. 1999 ima

	ponedeljaka	trinaestih
	utoraka	trinaestih
	sreda	trinaestih
	četvrtaka	trinaestih
*****	petaka	trinaestih
	subota	trinaestih
	nadalja	trinaestih

Ime i prezime _____

Adrese _____

Mesta _____

Pirati pred licem pravde

izvinite, nismo znali

Preuzimanje i preprodaja programa predstavlja sastavni, najživlji deo naše računarske stvarnosti i, otuda, omiljenu temu u mnogim računarskim razgovorima — od pošalice do naviјački obojenih rasprava „za i protiv". Autorski redovi računare pojačani su od nedavno jednim diplomiranim pravnikom, koji se redo prihvatio redakcijskog zadatka da zaviri u Zakon o autorskom pravu i izvidi kako ne ave to gledaju (kruta) pravne norme, koje obično nameju sluhe za „više interese". Šta očekuje Yu pirata ako izađe pred lice pravde?

Pre svega ko au to ‚pirati ? Korisnici računara baz aumnja, zneju da se oni bave nečim što nija dozvoljano Kako u oblasti računarenja nama ni brodove za pljačku ni aviona za otmicu ni dace za kidnapovanja, zaplatićemo es šta ja to što države zabranjuju a šta dopušta avojim građanima da rada u vazi računara? Mialim da ćemo sa svi složiti u jednom kada kažamo ‚pirat , pod tim podrazumavamo lice koja se bavi prasnimavanjem i prodajom kompjuterskih programa Pirati dakla nisu nikakvi razbojnici i ukoljice — devedeset odsto njih izgleda krajnja bezazlano moj lični pirat ima svega 12 godina i na izgleda nimalo krvoločno!

Domaća pravila ponašanja

Moramo međutim, prihvatiti još jednu činjanicu svaki tvorac naka ideja želi da tu ideju širi i saopšti drugim ljudima To je u interesu celog društva pa mu država u toma pomaže i štiti ga u njegovim nastojanjima lato tako u društvanom ja intarsu i da tvorac ideja ima i odradana ekonomska koristi od svog stvaralaštva — i država ima ekonomake koristi u vidu poreza Samim tim, kao u avom tako i u stvaraočevom interesu svake država propisuja odradana pravila ponašanja o toma na koji način sa ideja mogu širiti i skonomski iskorišćaveti Ukoliko sam tvorac ideja ili nako treće lice avesno ili nesvesno doda u sukob sa tim pravilima (na ponaša se onako keko je propisano) nijedna države neće mirno sedeti i gledati šta se dogada: već će naravno štiti svoj interes (u većini slučajava) i postarati se da prekršioca pojuri, stigne i adekvatno kazni

Da malo pracizziramo naka onsovne pojmove sveku tvorevinu iz oblasti stvaralaštva, bez obzira na vrstu, način i oblik izražavanja, zakon nazive *autorskim delom* Tvorac takve tvorevina naziva se *autor* a akup pravila ponašanja — prava i obaveza autora i prema autorima naziva — *autorsko pravo* Sve to raguliše *Zakon o autorskom pravu*, i još naki drugi zakoni Svim autorima država putem ovog sakona priznaja određena prava garantujući odgovarajuću pravnu zaštitu njihovim autorskim dalima Ukoliko sa ipak, desi da nako doda u sukob sa tim zakonima, verovali ili na ali on ja na udaru više dosta atrogih propisa ! višestruko ja odgovoran za njihovo nepoštovanja

Kako ja i program za računar tvorevina iz oblasti stvaralaštva izražen na specifičan način nama nikakva sumnja da i njaga možamo svrstati u kategoriju autorskih dala kao i da njegovom tvorcu pripada svoj-

atvo autora Samim tim, Zakon o autorskom pravu pruža pravnu zaštitu i programima za računara — keko programima naših državljana koji su objavljani u Jugoslaviji ili u inostranstvu ili koji još nisu objavljani tako i programima stranih državljana Što se tiče atranih programa, ukoliko su prvi put objavljeni kod nas (u atvarnosti malo verovatno ali sakon predvida i tu mogućnost) oni imaju istu pravnu zaštitu kao i programi naših državljana Ali ako su ti programi već objavljani u inostranstvu, situacija se jako razlikuja već prama toma kakve ja međunarodna obaveze prihvatila Jugoslavija na tom planu, i da li posotji takozvani ‚raciprocitet Kako međunarodni pravni odnosi niau predmat ovog članke poći ćemo od pratpostavka da naša zamlja u konkratnom slučaju priznaja stranom državljaninu pravo na pravnu zaštitu njagovog programa na tarritoriji Jugoslavija (štiti ga od besprsvnog ponašanja naših državljana)

I imovina i moral

Svako autorsko pravo pa tima i pravo autora programa za računara sadrži dve osnovne grupe prava autorska imovinska prava i autorska moralna prava.

Autorska imovinake prave sačinjavaju pravo (ovlašćenja) autora na iskorišćavanja avog dala (ukoliko to svoja ovlašćenja nija posabnim ugovorom praneo ne drugog kada ono pripada samo tom drugom) Zakon nabraja primara radi da sa autorako dalo (čitaj — program) može iskorišćavati od strane pravno ovlašćenog lica (čitaj — autore ili onoga na koga je autor praneo to ovlašćenja) na naki od sledećih načina objavljivanjam dala, praradivanjam dala, reprodukovanjam dala, umnožavanjem dala obradivanjam dala prikazivanjam dala izvodanjam dala njegovim prevođenjam ili ne drugi alićen način Potrebno ja ponovo naglasiti zakon navedana ovlašćenja priznaja i s k l j u č i v o autoru ili licu ne koga su ta ovlašćenja praneta ugovorom

Autorska moralna (lična) prave sačinjavaju prave autora da buda priznat i označen kao tvorac dala (svako od nas sa ponosi onim što ja atvorio), da se usprotivi svakom daformisanju i sakaćenju kao i da se usprotivi i avakoj upotrebi dala kojom bi sa vredali njegova čast i ugled lato tako i ovim pravima autora odgovaraju obaveze svih drugih da priznaju ta prava i da se uzdrže od bilo kakve radnja koja bi doiazila u sukob es navedanim pravilima

Pre nego što predamo na razaširjavanja pravnih zavrzlama koja se javljaju u vezi programa za kompjutera da konstatujemo nake činjanice

Kako ni autor ni njegova softverska kuća sa našim piratima na sklapaju ugovora kojim ih ovlašćuju na umnožavanja programa svi pirati se bave nazakonitim rabotama!

Svaki pirat ili haker koji „razbija programsku zaštitu naka igre ili drugog programa radi toga da bi omogućio njagovu distribuciju prako piratskih kanala, *prerađuja* u suštini program pa sa i on našao sa druga strana sakona!

Sveko ko sa bavi preprodajom programa bez valjanog pravnog cenova (bez odgovarajućeg ugovora se softvarskom kućom ili samim autorom), takođe ja, baz aumnja, zakračio na drugu stranu

Kako autor ima isključivo pravo da daja dozvolu za emitovanja svog dala putem radio-difuzne mraža postavlje se pitanja — znaju li urednici „Vantilatora da emitovanjam inostranih programa bez dozvola krše sakon?

Sa jakim začinom

Da bi ove pravna papazjanja bila malo ukusnija da navedamo odredbu zakona . Lice čije je autorsko pravo povređeno može sahtavati zaštitu avog prave i naknadu štate koja mu je povredom nanesena A pored toga sud prako koga autor buda ostvarveo zaštitu avojih prava može nareditі

1 Da sa presuda javno objavi na trošak povrediloca (obradovećete vaš džep jer oglasi niau baš jeftini a objavljana presuda ima format pola stranice Politike !)

2 Da sa povrediocu sabrani dalja povrada (posla takve sabrane ponovljena povreda će se uvek smatrati otežavajućom okolnošću što će imati velikog uticaja na težinu kazne koju aud bude izrekao)

3 Da sa predmati kojima je nanesena poveda autorskog prava uništa (da dobro sta pročitali) Znači moža da strada i vaš

računar sa kompletnom periferijskom opremom (kuku i lele) Ko ne veruje neka malo prelista zakon o autorskom pravu

Pirati, bolí li vas glava posle ovoga? Da još malo citiramo odredbe Zakona o autorskom pravu

„Ko pod svojim imenom ili pod imenom drugog objavi, prikaže izvede ili prenese tuđe autorsko delo ili dozvoli da se to učini, kazniće se za krivično delo zatvorom do tri godine"

„Ko deformiše skrati ili na drugi način menja tuđe autorsko delo kazniće se za krivično delo novčanom kaznom ili kaznom zatvora do šest meseci"

Ko bez dozvole autora ili drugog nosioca autorskog prava u slučajevima u kojima je takva dozvola potrebna po zakonu o autorskom pravu objavi preradi, reprodukuje prikaže izvede prenese ili prevede ili na drugi način iskoristi autorsko delo zaštićeno ovim zakonom kazniće se za krivično delo novčanom kaznom

„Ko u nameri pribavljanja materijalne odnosno imovinske koristi stavi u promet i primerke autorskog dela za koja zna da su neovlašćeno reprodukovani ili umnoženi ili ko takve primerke javno izloži ili ih prenese putem radio-difuzije ili na drugi način, kazniće se za krivično delo novčanom kaznom

Da veselje bude veće da napomenemo da ukoliko neko od navedenih krivičnih dela učini organizacija udruženog rada ona odgovara još i za prekršaj a kazna je prava , eltnica" — trideset sterih miliona!

Kao prvo, zakon kaže da se za navedena krivična dela gonjenje preduzima po privatnoj tužbi Šta to u stvari, znači? U našem pravnom sistemu postoje dve vrste krivičnih dela kod jednih, država preduzima krivično gonjenja odmah čim sazna za delo ili učinioca (u pitanju su teža krivična dela koja su po društvo opasnije) Za koju vrstu država neće reagovati čak i ako sazna za dela ili učinioca sve dok neko na podigne tužbu Ili podnese krivičnu prijavu protiv učinioca (obično onaj ko je tim delom oštećen) Radi se pogadate o krivičnim delima koja se gone po privatnoj tužbi Pošto krivična dela koja emo malopre citirali spadaju u ovu drugu grupu, možete mirno da spavate sve dok se neka strane ili domaće softverska kuće (i to je moguće!) na naoštri protiv vas zbog toga što neovlašćeno prodajete ili nešto drugo radite sa njenim programima i na počne da vas progenja A kako naši pirati vole da pametuju i da se u programima koje rastuvaju potpisuju punim imenom, prezimenom i telefonskim brojem, pretpostavljate da zainteresovanoj softverskoj kući ne bi predstavljalo problam da nađa dotičnog i da ga tuži sudu Međutim mala je varovatnoća da će do progenjanja i doći, jer bi Iroškovi sudskog postupka za onoga ko ge pokrene bili u velikoj nesrazmeri sa onim što bi uspešnim okončanjem postupka zaista i dobio pe se takve robota, u sadašnjim uslovima, jednostavno — ne isplati

All, ipak pretpostavimo da ima upornih na ovom belom svetu i da sa neka kompenzija Ipak rešite da vas progoni do kraja Uostalom pojavili su se i prvi domaći izdavači softvera Postoja dve mogućnosti — bićete tuženi ili za naknadu štete ili za krivično delo Da kraenemo redom

Nastavak u sledećem izreju

Nenad Mitrović

25/izvinite, nismo znali

Hakerski vodič kroz Beograd

u potrazi za izgubljenim vremenom

Osnovna želja kojom smo se rukovodili kada smo planirali ovaj prilog bila je da pomognemo potencijalnom kupcu računara da ostane u Beogradu, prođe što jeftinije i kupi najkvalitetniji: robu i spo računa, ali nije išlo! Ovaj teket, koji je sede pred vema, esmo je biedi prirepak onoga što je trebelo da bude. hakeraki vodič kroz Beograd. Ali, podimo redom — Iae ovo melo podeteke moći ćete da steknete nekakvu aliku o etanju računeratva kod nas. .

Prvi deo krećemo u akciju

Krenuli smo od železničke stanice kao mesta odekle bi potencijalni kupac iz unutrašnjosti otpočeo svoje beogradsko putešestvije U njenoj okolini računara ni za lek! Samo nekoliko komisiona i knjižara sa alakričnim pisećim mašinama All stop! U Karađorđevoj ulici — to je ona kojom ide Iramvaj za Kalemegdan — nailazimo na veliku reklamu za komodor" Ulazimo Lepo uređena unutrešnjost gomila opreme (uvoz) ploče kasete uređaji kablovi sintisajzeri gitare I evo RAČUNARI! Naravno – komodor Tu su svi — PC 10 C128 C64 (i novi i stari dizajn) kasetaši diskovi monitori džojstici, jednom rečju praznik za oči!

Iz razgovora sa ljubaznim prodavcem (ovda su svi ljubazni) Darkom Koslćem saznajemo da je –Max Profi Shop prodavnica tehnička robe koje radi pod okriljem Mataisarvisa I bavi se prodajom računara (i na samo njih) preko konsignacije Dobili smo i cenovnik sa devizninm iznosom i dinarskim delom Da vidimo kako to izgleda

C	64	235		1541	250		1530	32		1801	270		803	155
C	128	330		1570	265		1531	25		1901	380			
PC	10	1720		1571	315									
računari				diskovi			kasetaši			monitori			štampači	

U tabeli vidlia pojadini tip računera ili oprema i pored njega podvučen podatak o ceni u davizame (dolari!) e ne tu cenu Irebe dodati i oko 65% u dinarima na ime poreza

U radnji možete kupiti i originalni džojstik po ceni od 5 5 dolara i diskale za oko 1 5 dolara po komadu Zanimljivo je da imaju i kartridže sa igrama koji košleju oko 5—15 dolara što je straobalno skupo u odnosu na naše drage pirate (Gornjim cenama pridodajte onih nazbežnih 65% u dinarima)

Raspitali smo se za mahanizam prodaje Rečeno nam je da robu dobijaju od – Konima iz Maribora i da sva isporuke zavise od njega i njegovih mesečnih isporuka Za sve informacija možete zvati broj 011 (624 -927) i saznati pojedinosti Inače kade roba stigne radnja sa ispuni kupcima i po pedesetak njih pa sve tići na pijacu ili mesarnicu Radno vreme prodavnice je radnim danom od 7—19 časova subotom od 8—15 časova a adresa Karadordeva 65 all ne možete da omanete jar se velika reklama vidi izdaleka

Slobodno svratita jar vam je to možda jedine prilike da vidita i opipate — to se ovda ne brani) C 128 ili PC 10 Na rastanku, u poverenju nam je rečeno da Max Profi Shop pregovara sa Amatradom oko konsignacione prodaja u Beogradu! Držimo im palčeve!

Drugi deo melodrema u šopu

Sledeći na našem putu je bio neravno kompjutar šop iliti Compular Shop radnja na koju pomisli 80% vlasnika računara kade krena da pazari nešto od opreme po Beogradu Da ti s pravom vidačemo

Adresa prodavnice je Maršala Tita 48 ali je to, u stvari ulica Generala Ždanova Radnja je lepo uređena ali se nigde na vidu računari samo knjige ovaj ah da u jednom ćošku sede dva mladića sa nekakvim računarom Da im pričamo! Dogadaji dalje taku kao u nekom skaču pe ih tako i opisujemo

Mi — Dobar dan!
Oni (ćute). (U stvari, jedan od njih dvojice lupa po tastaturi .Amstrada I izbacuje iz usta sistemska poruka kao Integer out of range Syntax error No data itd dok drugi priča sa nama)
Mi — Dobar dan! (opet)
Oni — Dobar dan izvolite
Mi — Interesuje nas amstrad videli smo oglase u novinama šta prodaje te ovde može li da se vidi cenovnik može li da se proba nešto
Oni — Da da to je bila greška mi ne vršimo prodaju amstrada to je palo u vodu ovo što imamo
Mi — Da li možemo da isprobamo recimo spektrum sa epsonom
Oni — Ne, žao nam je ali to na može osim ako kupujete
Mi — Što ste zagradili opremu stolom? Mislite da će vas neko pokrasti? (imao sam nameru da iznesem , epson pod miškom)
Oni — (ćuta onaj drugi seda više na izbacuja sistemska poruka već se divi kolor-grafici amstrada u MODE 1 koju je skrpio sa dva FOR--NEXT petlje)
Mi (iziazimo razočarani)

Koliko amo mogli da saznamo u , Computer Shopu se snabdevaju samo preduzeća se pozamašnim računima jar da li biste vi platili tri-četiri puta skuplje spektrum ili dali za domaći printer oko 45 miliona da li na govorimo o epsonu koji sa približava stratosferi

Na žalost a možda i srećom jedina stvar koja vredi u „Computer Shopu ceim računara koji su pravredni su knjige Ali o toma

Idamo dalje tu su i komisioni Ima ih puno i nije ih lako sva obići za kratko vreme a i nemaju bogzna šta da ponuda Prosačne cena računare (ako ih imaju) su spektrum — 90 000 do 130 000 ,komodor 64 oko 200 000 din ,C–16 oko 120 000 din, dataset za komodor oko 30 000 i to kopija disk-drajv 1541 190 000 din našli smo i na video-igre koje koštaju 60 000 i šah-kompjuter od 60 000 din ali to nija cilj ovog napisa

Sva u svemu žalosno Izbor je loš a računari su drastično preceniani Uostalom komisioni i nisu mesta gde se kupuje prava roba To dokazuje i komplet od 10 diskata (5,25) na kojima namatijivo plža „IBM i koje koštaju oko 25 000 din! Stop

Po robnim kućama i prodavnicama bira--opreme vlada sivilo, tak ponegde razbijano kojekakvim ludorijama naše trgovačke umešnosti idamo redom

Prodavnica Prosveta M Tita 6 inače odlično snabdevana papirnica Ima (ko zna kako?) ZX Interface 1 i ZX Microdrive Svaki košta po 54 500 din Na pitanje o poreklu roba i (eventualnim) budućim isporukama dobili smo odgovor u stilu ko zna?

Treći deo bežimo napolje

Prosveta u Čika Ljubinoj 1a odnakud ima računara! U razgovoru sa simpatičnom prodavačicom saznali smo da računare otkupljuju od komisona (???), a onda ih prodaju Divno! Imaju spektrum za 80 000 din, C–16 (opet!) za 70 000 din bez dataseta i ispravljača, a se njima oko 160 000 din I oni na znaju šta će da dobiju i koliko Lepo Najmešniji deo ovog istraživanja po asfaltnoj džungla na asfaltu na temperaturi od oko 32°C u hledu zbio se u prodavnici ,Elektrotehna u ulici M Tita (ona veća prodavnica) Tu imaju računar Oric Nova 64 ' I prodaju ga za lepa pare — 161 860 din Teško ja reći odakle im baš ta cifra znamo samo da prodavac nija znao ama baš ništa sa njim i oko njega non-stop tvrdeći da uz računar ida i nekakav medijum za spremanja podataka! Kada smo ga zapitali šta je to izvedio ja naku crveno-crnu kutiju ispod tezga I trijumfalno nam ja poturio pod nos Posle dva minuta I blagog šoka koji je uiledio ustanovili smo da ja ta kutija u stvari ispravljač računare, koji je neerećnim alučajem u istoj kutiji u kojoj bi trabalo da se nalazi 3-inčni disk-drajv!!! Naravno drajva ni od korova dok ja prodavac odbio da sasluša nesle predavanja o vrstama drajvova I njihovom izgledu Svejedno u prospektu koji se dobija od proizvođača, piše da posloji i drajv i da će mu cena biti oko 460 000 din I Brzo brzo bežimo napolje!! Što se tiče TV ili monitora, imate šarolik izbor mini-TV koja proizvode Ei ili RIZ a cena im se kreće od 60 000 do 90.000 din U „Šumadiji , Knez Mihailova 26 u prizamiju imate dobru papirnicu, dok na spretu vrvi od računara i pisaćih mašina U razgovoru se prodavcima saznali smo da se Šumadija bavi uvozom i zastupništvom mnogih firmi i da mogu da, ako se pokaže interesovanje, uvoze pojedina artikla koji su potrebni računardžijama Tu spadaju diskete kasete papir, traka za printera, itd

Što se tiča računare, možete da pogladate „lolu 8 po ceni od 140 000/160 000 175 000 za varzija od po 16/24/32 K Od pribdra trenutno su imali diskata od 8 inče (4000 din-) 5,25 inča (3500), 3.5 inča (5250)! Tu su i streamer-traka kapacitata 700K za oko 3400 din Šumadija Inače prodaja široku gamu ozbiljnih mašina i sistema za obradu podataka, što će zanimati firma, dok su cena za pojedince prevelika Napomenuto nam ja da se ako buda interesenata može kod njih potražiti i diskate od 3 inča („amatred) i ava vreta traka za printera Potrebno je da donesete svoju traku i da provarite da li imaju odgovarajuću

Da rezimiramo, Beograd nija mesto u koje biste sa mogli uputiti radi kupovina računara Ako amo u ovom tekstu i propustili ponaku prodavnicu, zastupnika ili mala oglase, to je zbog toga što ni oni na bi popravili ovako lošu sliku To ucetalom dokazuja i jedan par pocepanih patika istrošenih lunjanjem po vrelim ulicema Beograda, a ava u (uzaludnoj) potrazi za računarima Minhen je izgleda bliža

Darko Stanojević

Šta ima novo

Modesti Blejz na „meku

Imali ste samo dva pasija u životu — crtanja atripova i računare Tu ja odmah bio i problem Na samo što niste imali vremena i za jedno i za drugo, nego ste i sve pare koje ste nekako zaredili crtajući stripove poarčili na računareku opremu ili još gore ava pare koje ste zaradili preprodajom programa potrošili ate na atripovaki repromaterijal Evo rešenja Program se zove Comic Works i služi za crtanja atripova na „meku Pomoću njega možeta crtati, bojiti, uređivati tekat i, naravno mešati crtež i tekst na bilo kom dalu atranice Programvam još daje izvestan broj dodatnih elemenata kao što su balončići za govor, koji se mogu slobodno rasporedivati po atrani Program potražite od firme Mindacape Inc 3444 Dunder Rd. Northbrook, Illionois Usa Comic Works košta 79,95 dolare (B Đ)

Keko podmladiti traku

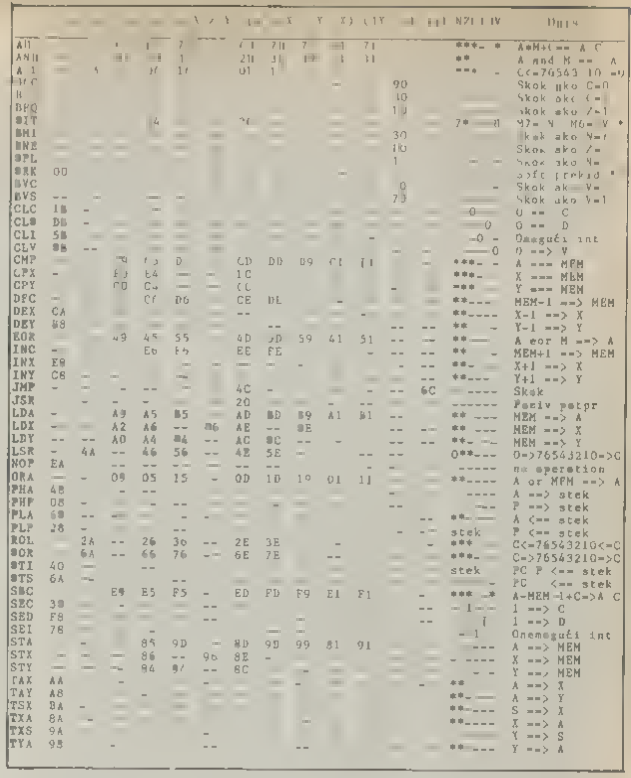
Ako imate printer sigurno ste primetili da ta draga sprevica guta kasete sa trekama se zastrašujućom proždrljivošću i katastrofalnim posledicama po vaše ionako minimalna količina deviza Englezi sličnе probleme imaju odavno, pa se zato naša firma koja je smislila kako da im pomogne Firme Aladdink iz Berwikšajera za trećinu cena nove trečne kasete cevežava, to jest boji atare Bilo bi lepo kad bi se i kod nas neko setio da tako nešto radi (B Đ)

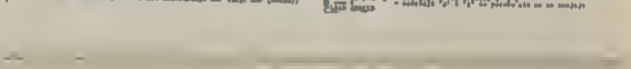
Qualitas za kvalitet

Ako imate ZX spektrum i neki od starijih „epson printare, a želite da štampate u NLQ modu — nema problema! Idite do prve „Computer-shop tabla (na kod nse) i kupite program „Qualitas Taj program omogućava visenicima ,spektruma koji pišu sa takat-procesorom „Tasword Two , da imaju onu mogućnost ispisa na papir koja se danas toliko ceni — NLQ

Naima, program koristi dva proiseka glava printare i mikro pomaranja vaijka Dalja može sa koristiti proportionalni i equal-space ispis sa pet vreta slova koja se mogu izmeniti pomoću editore slove koji se prilaže uz program Progrem radi se avim „EPSON RX-80 printarima i njegovim kompatibilicima uh

Verzija za Tasword Two košta 7 95 funti a za Tasword Tres će koštati (kada se pojavi) oko 8 95 funti





Dejan Ristanović

6502
mašinac za početnike

Kada se u nakom od domaćih kompjutemkih časopisa pomene škola mašinskog programiranja, zna se koma je namenjena — vlasnicima računara opremljenih mikroprocesorom Z80. Korisnici komodom i „apla BBC-ja „electrona i mnogih drugih popularnih kompjutera se svakako osećaju zapostavljeno — zar je njihov 6502 toliko loš da o njemu na vredi pisati?

Računari će prakinuti „zavaru ćutanja i pokušati da, kroz ovaj umetak približe mikroprocesor 6502 početnicima

Procesor 6502 je relativno star i izuzetno popularan proizvod firma MOS Technology u proteklih je desetak godina oko njega dizajniran vajki broj veoma uspešnih kompjutara bezbroj samogradnji i određena količina kontrolera inteligentnih interfejsa i slična oprame. Razlog za popularnost 6502 su svakako njegova karakteristika brzina i jednostavnost programiranja. 6502 doduše ima manja registara i instrukcija nego Z80 ali vza ovaj nedostatak u ovom tranutku može samo obradovati — ako postoji manje instrukcija brže ćete ih upoznati i lakše zapamtiti. Zanimljivo je primatiti da se u poslednja vrema u razvoju mikroprocesora zapaža težnja da se broj instrukcija smanjuje a brzina povećeva — 6502 bi se možda mogao smatrati za jednog od rodonačelnika modarne RISC arhijektum. Videćemo da je njegov set instrukcija simanjičan i logičan da je na njamu lako progmmirati i da sa programi koje napišete relativne lako testiraju i modifikuju

Iako varujamo da ste veoma zainteresovani da počnete da programirate ne možamo bez odmah da predemo na mikroprocesor 6502 — treba se najpm upoznati se osnovnim pojmovima. Obzirom na obim ovog umetka i činjanicu da smo želeli da ge dopunimo pristojnim brojem tabela i primara namamo previše prostora za ovaj uvod Tu ja na sreću naš umetak „Z80 — mašinac za početnika iz Računara 15 Teško nam je latina da vas upućujamo na literaturu posvećenu omrženom Z80 Ipak uvodna poglavlja našeg umetka su pisana tako da budu saavim opšta i da sa odnose na razne mikroprocesora pa i na 6502 U okviru ovog poglavlja ćemo dakle pobrojati činjanice koja trabe da znate kako biste mogli da se upustite u programiranja na mašincu ako vam neka od jih činjenica nija baš jasna pogledaja dalako opširnija objašnjenje iz Računara 15

Mikroprocesor ja centmmi deo svih kućnih računara i različita druge oprama koju nazivamo inteligentnom Ovaj termin označava čip koji može da se programim za procesiranja tj obradu nekakvih podataka Moderni kućni računari u sebi sadrže više čipova koj bi sa mogli svrstajl pod našu definiciju mikroprocesora ukoliko na primer vaš računar povezuje se drakom kontrolu ovoga uređaja će obavljati disk kontrolar ji specijalizovani mikroprocesor Ako pročitaja da vaš kompjuter ima video kontrolar znaćete da se radi o još jednom mikroprocesoru koji se brina za generisanja slika sličnih primam ima još dosta Mi se, međutim, nećemo baviti specijalizovanim kontrolarima već mikroprocesorom 6502 (na zabrinjavajte se ako je vaš aiicijumski ljubimac opremljen mikroprocesorom 6510, 8502 ili nekim aličnim — ovi se čipovi programimiju kao 6502) koji se nalazi u srcu nekolikо danas veoma popularnih modela kućnih računara Centrsini mikroprocesor je zaduđen za izvršavanja mašinskih programa (dok na primar izvršavaja bejzik program traja izvršavanja bajzik interpmtera koji je upisan u ROM vašeg kompjutera) i za koordinaciju rada ostalih specijalizovanih kontrolera ugrađenih u računar Korisnik obično nema nikakvih mogućnosti (a ni potreba) da komunicira sa bilo kojim hardvamkim modulom svoga računara osim posredstvom centralnog mikroprocesora

Procesor 6502 je kao i Z80 **osmobitni** mikroprocesor, što znači de je najveća memorija koju možete da mu prikljućite 64 kilobajta. Ova je memorija podeljena na *ćelije* koje ćesto pomalo nepravilno nazivamo *bajtovima* — pročitajte Računam 15 pa ćete videti u čemu je razlika Memorijska ćelija označavamo *adresama* ćelija 0 ćelija 1 i tako dalje — sve do ćelije broj 65536 Svaka je ćelija se svoje strana podaijana na osam *bita* svaki bit može de ima vraćnost 1 ili 0

Operativna ...

Osam bita ili jedan bajt ja najmanja količina infermacija kojoj mikroprocesor može da pristupi To snači da će na primar naredba LOAD A 20000 doveti u akumulator (jedan od registam procesora nija bitno ske ovaj jarmin još na rsaumaja) sadržaj memorijska ćelija čija je adresa

20000 i te svih osam bita koja sadrži ta ćelija. Ukoliko nam je potrebno da pojedinačne menjamo neke od tih bitova moračemo da dovedemo čitavu ćeliju u akumulator, modifikujamo je a zatim je celu vratimo u memoriju

Memorija se kao što smo videli sastoji od ćelija koje su numerisane po svojim rednim brojevima Pretpostavimo da našem računaru treba da naredimo da u ćeliju čiji je broj 100 upiše zbir sadržaja memorijskih ćelija čije su adrese 101 i 102. Bilo bi logično da se za takvu naredbu upotrebi oznaka

ADD 101, 102, 100

ADD je skraćenica za naredbu „sabari" i prate je tri adrese adrese memorijalne ćelije u kojoj je prvi sabirak, adresa drugog sabirka i adresa ćelije u koju treba staviti rezultat. Pokazalo se nezgodnim da se u okviru svake naredbe pominju po tri adrese pa su se ljudi dosetili registrima — pogledajmo ekvivalent gornja naredbe

LOAD A (101)	Prenesi u akumulator sadržaj ćelije 101
ADD A (102)	Dodaj ne sadržaj akumulatore sadržaj ćelije 102
	rezultat sabiranja smesti u akumulator
LOAD (100) A	Prenesi sadržaj akumulatora u ćeliju 100

Vidimo da umesto jedne naredbe sada moramo da iskoristimo tri ali da zato svaku od njih prati samo po jedna adresa. Uz adresu se međutim stalno pominje nekakvo slovo A koja predstavlje ime jednog registra procesora.

Osim operativne memorije koja se sastoji od šezdesetak hiljada ćelija 6502 ime i nekolike bajtova svoje privatne memorije koji su grupisani u registre tip-flopovi koji sačinjavaju te registre se nalaze unutar samoga mikroprocesora. U jednom od sledećih poglavlja ćemo detaljno upoznati registre 6502

. . . i privatna memorija

Sledeći termin koji treba da upoznamo je *flag*. Pretpostavljajući da ste razumeli šta je registar reči ćemo da je flag jednobitni registar. Ovo bar na prvi pogled dolazi u kontradikciju sa našom rečenicom tvrdnjom da mikroprocesor može da pristupa samo čitavim bajtovima memorije ali ja se kontradikcija samo prividna flagovi nisu deo memorije već deo samog procesora i imaju veliku ulogu za njegov normalan rad. Pretpostavimo da treba u ćeliju 100 upisati broj 265 ukoliko se u toj ćeliji ranije nalazila nula ili broj nula ukoliko se u toj ćeliji ranije nalazilo bilo šta drugo. Posmatrajmo program se slika 1 koji bi mogao da obavi tu operaciju

```
Au      A #00       Stavi sadržaj ćelije 100 u akuaulator
        CMPE A ,C   Uporedi sadržaj akumulatora sa nulom
Pa\     EQ NULA     ; kolika je jednak (EQ) ; idi do zeate
                    u programu koje se zave  NULA
Au      A #0        Broj nulu u akumulator
T P     A 100       ; a zatim i u ćeliju 100
        Kraj rada

L A                 Ovde se dolazi ako je sadržaj ćelije 100
                    na pačerku bio 0
Al      A #255      Broj „55 u akumulator
T pc    A 100       ; a zatim u ćeliju 100
Ak                  Kraj rada
```

slika 1

Vidimo da je instrukcija COMPARE A #0 bila iskorišćena da se testira sadržaj akumulatora, a zatim smo upotrebili naredbu JUMP EQ NULA koja je predstavljala skok samo ako je odgovor na pitanja koje je postavlja prethodne naredba bio potvrdan. Kako bi naredba JUMP znala kakav je rezultat izvršavanja prethodna naredba? Jedino tako što je naredba COMPARE A #0 postavila naki flag mikroprocesore u stanja 0 ili 1. Taj flag će biti u stanju jedan ako je odgovor na pitanja „da li je sadržaj akumulatore jednak nuli" potvrdan a nula ukoliko je odrečan. Sada se naredba JUMP EQ NULA može protumačiti rečima „Idi na deo programa označen sa NULA ukoliko je flag jednak jedinici (ili kako se to obično kaže ukoliko je flag *setovan*) a produži se izvršavanjem programa ako nije

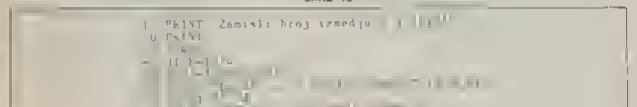
Procesor 6502 ima više flagova kojima su dodeljene specijalna značenja. O njima ćemo jasno, opširno govoriti docnije za sada je bitno da zamislimo flag kao zastavicu koje se podiže ako je odgovor na neka pitanje potvrdan a spušta ako je odrečan stanja te zastavice docnije možemo da testiramo i da menjamo tok izvršavanja programa u zavisnosti od njega

Stek

Za kraj smo ostavili pojam *steka*. Obzirom da svaki mikroprocesor ima relativno malo registara često će nam biti potrebne da neka od njih privremena oslobodimo za obavljanja neka operacije

```
3099                            ; treba ispisati na ekran
3099                            ; o sadržajeka ćelije 'ucr
3099 8E 2A      STA ecr_low
309B A0 BB 31 LBA proba_hlph
309E 8E 2B      STA ecr_high
30A0 20 1F 99 JSR eeprna        ; Ispisivanje pakesaja na ekranu
30A3 A9 20      LDA #ASC " "    ; Ispisuja se jedno blanko
30A5 20 EB FF JSR eewrch
30A8 60         RTS             ; Povratak u glavni program
30A9 ,          tekst
30A9            ; tekstovi poruka
30A9 0A         EQUB 10
30AA 5A 61 6D EQUS "Zamisli broj izmedju 0 i 1023"
30C7 00         EQUB 13
30C8 0A         EQUB 10
30C9 61 20 6A EQUS "a ja cu ga pogoditi!"
30DO 0D         EQUB 13
30DE 0A         EQUB 10
30DF 0A         EQUB 10
30E0 50 6F 73 EQUS "Posle svakog tog pakesaja pritisni
3103 0B         EQUB 13
3104 0A         EQUB 10
3105 20 2D 20 EQUS " - M ako je tvaj broj manji."
3121 0D         EQUB 13
3122 0A         EQUB 10
3123 20 20 20 EQUS "  - V ako je tvoj broj veci
313E 00         EQUB 13
313F 0A         EQUB 10
3140 20 20 20 EQUS "   P ako mi pogodio."
3155 00         EQUB 13
3156 0A         EQUB 10
3157 0A         EQUB 10
3158 00         EQUB 0
3159            .pracpt
3159 53 74 61 EQUS "Sta kazes za broj
3168 00         EQUB 0
3169            .pobede
316C 50 6F 67 EQUS "Pogodio sam (apat)!"
317F 0B         EQUB 13
3180 0A         EQUB 10
3181 0A         EQUB 10
3182 00         EQUB 0
3183            . Promenljivei
3183 00         .donja_low EQUB 0
3184 00         .donja_high EQUB 0
3185 00         .gornja_low EQUB 0
3186 00         .gornja_high EQUB 0
3187 00         .proba_low EQUB 0
3188 00         .proba_high EQUB 0
```

slika 43



slika 44

```
3000              hi_iew_gese
3000 A2 00        LDX #0                  Ispisivanje upustetva
3002              upustetve
3002 BD A9 30     LDA tekst X
3005 F0 07        BEQ igre
3007 20 EE FF     JSR oswrch
300A E8           INX
300B 4C 02 30     JMP uputstva
300E              igre                    Broj je izmedju 0
300E A9 00        LDA #0
3010 8D 83 31     STA donje_low
3013 8D 84 31     STA donje_high
3016 8D 85 31     STA gornje_low
3019 A9 04        LDA #4                  1 1024 = &40
301B 8D 86 31     STA gornje_hiph
301E              pokmoej
301E 18           CLC
301F AD 83 31     LDA donje_low           Racuna se (donje+gornje)/2
3022 6D 85 31     ADC gornje_low
3025 8D 87 31     STA proba_low
3028 AD 84 31     LDA donje_high
302B 6D 86 31     ADC gornje_high
302E 8D 88 31     STA proba_high
3031 4E 88 31     LSR proba_high
3034 6E 87 31     ROR proba_low
3037              pokusajl
3037 20 4B 30     JSR ispisi_proba        Na ekranu se ispisuje 'proba'
303A 20 E0 FF     JSR osrdch              Karlenik treba da pritisne tester
303D 20 EB FF     JSR oswrch              Slovo koje je pritisnuto se ispisuje
3040                                      i na ekranu radi provere
3040 48           PHA
3041 A9 00        LDA #&0D                Prelazak u novi red
3043 20 EE FF     JSR oswrch
3046 A9 0A        LDA #&A
3048 20 EE FF     JSR oswrch
304B 68           PLA                     Ispituje se slovo koje
304C                                      je korisnik otkucao
304C C9 50        CMP #ASC "P"
304E F0 29        BEQ pogodio
3050 C9 4D        CMP #ASC "M"
3052 F0 16        BEQ manji
3054 C9 56        CMP #ASC "V"
3056 F0 03        BEQ veci                Ako nije P, M ili V pita ponovo
3058              veci
3058 AD 87 31     LDA proba_low           Donja = proba
305B 8D 83 31     STA donje_low
305E AD 88 31     LDA proba_high
3061 8D 84 31     STA donje_high
3064 4C 1E 30     JMP pokmsaj
3067              manji
3067 AD 87 31     LDA proba_low           Gornja = proba
306A 8D 85 31     STA gornje_low
306D AD 88 31     LDA proba_high
3070 8D 86 31     STA gornje_high
3073 4C 1E 30     JMP pokmsaj
3076              pogodio
3076 A2 00        LDX #0                  Ispisivanje trijumfalne poruke
3078              ispis_kraja
3078 BD 8C 31     LDA pobeda X
307B F0 07        BEQ kraj
307D 20 EE FF     JSR oswrch
3080 E8           INX
3081 4C 78 30     JMP ispis_kraja
3084              kraj
3084 60           RTS                     Kraj programa
3085              ispisi_proba
3085 A2 00        LDX #0                  Ispisuje se pokusaj kompjutera
3087              ispisi_prompt
3087 BD 59 31     LDA prompt X
308A F0 07        BEQ kraj_prompta
308C 20 EE FF     JSR oswrch
308F E8           INX
3090 4C 87 30     JMP ispisi_prompta
3093              kraj_prompta
3093 AD B7 30     LDA ispisi_proba+2      Ispisuje se proba
```

da bismo po njenom završetku, razmotrili njihov raniji sadržaj. U takvoj situaciji bismo mogli da prepišemo sadržaj log. registra u memoriju, ali bi se takvo smeštanje pokazalo neracionalnim. Pogledajmo zato jednu od varijanti upotrebe steka.

```
PUSH A          Registar A se čuva na steku
                . Deo programa koji koristi akumulator
PULL A          . Obnavlja sadržaj akumulatora.
```

Sadržaj registra A je i ovde prepisivan u neku memorijsku ćeliju, ali sada mi nismo morali da razmišljamo o njenoj adresi: mikroprocesor je opremljen registrom S (*Stack pointer* ili *pokazivač steka*) koji sadrži adresu memorijske ćelije koju treba popuniti brojem koji se šalje na stek. sadržaj ovog registra se automatski povećava ili smanjuje po svakoj operaciji sa stekom, tako da primanom PULL instrukcije uvek dobijamo podatak koji smo „zapamtili" primenom prethodne PUSH naredbe. Stek se, osim toga, intenzivno koristi za realizaciju potprograma i prekida i posvetićemo mu dužnu pažnju u jednom od sledećih poglavlja.

Predstavljanje podataka

Brojevi mogu da se prikazuju u raznoraznim sistemima — dekadnom, oktalnom, heksadekadnom, binarnom... Posmatrajmo na primer broj 1986. On je iskazan u pozicionom dekadnom sistemu. Ovo *dekadnom* potiče od toga što je baza sistema broj 10, što znači da oznaka 1986 predstavlja zbir sa slika 2

```
1986 = 6*10^0 + 8*10^1 + 9*10^2 + 1*10^3
     = 6*1    + 8*10   + 9*100  + 1*1000
     = 6      + 80     + 900    + 1000
```

slika 2

Što se broja 100101100 dâ... on je iskazan u pozicionom binarnom sistemu. Ovo *binarnom* potiče od toga što je baza sistema broj 2, što znači da oznaka 100101100 predstavlja zbir sa slike 3

```
  100101100
=0*2^0 + 0*2^1 + 1*2^2 + 1*2^3 + 0*2^4 + 1*2^5 + 0*2^6 + 0*2^7 + 1*2^8
=0*1   + 0*2   + 1*4   + 1*8   + 0*16  + 1*32  + 0*64  + 0*128 + 1*256
=0     + 0     + 4     + 8     + 0     + 32    + 0     + 0     + 256
=300
```

slika 3

Vidimo da binarni broj 100101100 i dekadni broj 300 predstavljaju istu stvar izraženu na dva različita načina. Prvi način predstavljanja je pogodniji za računar a drugi za čoveka i postoje postupci kojima se brojevi iz jednog sistema prevode u drugi. Ovi su postupci detaljno objašnjeni u „Računarima" 15.

Za mašinsko je programiranje posebno značajan *heksadekadni* sistem sa osnovom 16. Ovakvo predstavljanje podataka, osim cifara 0—9 zahteva i cifre A, B, C, D, E i F pa neki heksadekadni broj može da glasi &12A4 (& uvek označava heksadekadni broj). Konvertovanja heksadekadnih brojeva u binarne i binarnih u heksadekadne je uzurpno jednostavno, pa nam potrebna nikakve tabele programi ili algoritmi. Pokušajte dakle da zapamtite binarne predstave heksadekadnih cifara, se slika 4

Dekadno	Binarno	Heksadekadno
0	0000	& 0
1	0001	& 1
2	0010	& 2
3	0011	& 3
4	0100	& 4
5	0101	& 5
6	0110	& 6
7	0111	& 7
8	1000	& 8
9	1001	& 9
10	1010	& A
11	1011	& B
12	1100	& C
13	1101	& D
14	1110	& E
15	1111	& F

slika 4

Operacije sa binarnim brojevima

Sa binarnim i heksadekadnim se brojevima računa kao i sa običnim — treba samo znati rutinu i zamenih standardnu tablicu sabiranje i množenja novam — u heksadekadnoj je tablici na primer 8+3=C dok je u binarnoj 1+1=10

Mali problem predstavljaju negativni brojevi — za računar nije mnogo prihvatljivo memorisanje minusa iz —12 Ako pretpostavimo da se svaki broj sastoji u jedan bajt memorije mogli bismo da se dogovorimo da prvi (ili poslednji eko kao što je uobičajeno bitove brojite zdesna) bit tog bajta u sebi kodira znak tako da će se broj 7 pamtiti kao 00000111 a broj —7 kao 10000111 Ovakve predstavljenje brojeva se međutim pokazalo nepogodnim za mnoge primene pa je izmišljeno nešto bolje — pamćenja komplementa broja *Prvim komplementom* broja 7 (00000111) nazivamo binarni broj koji dobijamo kada umesto svake nule u tom broju stavimo jedinicu a umesto svake jedinice nulu —00000111 tako postaje 11111000 Dodajući jedan na ovaj broj dobijamo 11111001 što je *drugi* ili *potpuni komplement* broja 7 I ujedno zgodan način da se predstavi broj —7 Sami se uverite da se računanjem sa brojevima u potpunom komplementu dobijaju korektni rezultati Sve u svemu

1 U memorijsku ćeliju mogu da se smeštaju označeni ili neoznačeni brojevi U jedan bajt (osam bita) mogu da se smeste označeni brojevi između —128 i +127 ili neoznačeni brojevi između 0 i 255

2 Pri sabiranju neoznačenih brojeva može da se pojavi prenos iz najvišeg razreda koji nazivamo *carry* Pojava ovoga prenosa pri računanju implicira postojanje greške tj pokušaj da se saberu dva broja čiji je zbir veći od 255

3 Pri sabiranju označenih brojeva *carry* treba ignorisati Moguće je međutim da se pojavi prekoračenje tj greška u računu koju nazivamo *overflow* Mikroprocesor 6502 omogućavaju programeru da proveri postojanje greške (koja čini rezultat u akumulatoru besmislenim) i preduzme neku akciju ukoliko je ona nastupila.

Logičke operacije

Programeri koji svoje ramek-dele pišu isključivo na višim programskim jezicima možda smatraju da su sabiranje oduzimanje množenje, deljenje i eventualno stepenovanje jedine operacije koje računar može da obavi Ukoliko se međutim upoznate sa nekim ko u životu piše jedino mašinske programe verovatno ćete saznati de računari umeju da sabiraju (eventualno i da oduzimaju) da se sa množenjem alebo enalaze da im deljenje predstavlje veliki problem ali de su im operacije čudnih imena *konjukcija, disjunkcija, negacija* i *šiftovanje* najjača strana To je razlog što ovim operacijeme koje su vam verovatno alebo poznate posvećujemo čitavo poglavlje

Konjukcija je latinski naziv za logičku i funkciju dok je *disjunkcija* logičko ili — ova su operacije definisana tabelame sa slika 5 detaljna primera ovih operacija ćete takođe pronaći u Računarne i5 Konjukcije se koriste za brisanje pojedinih bitova nekog od registara dok će nam disjunkcije poslužiti za njihovo setovanje, dacnije ćeme naučiti mnogo više o ovim operacijame

X	Y	X and Y
0	0	0
0	1	0
1	0	0
1	1	1

X	Y	X or Y
0	0	0
0	1	1
1	0	1
1	1	1

X	not X
0	1
1	0

slika 5

Osim konjunkcije i disjunkcije koristićemo i logičku negaciju (NOT) čije je istinitosne tabele takođe data na slici 5 Primena ove operacije je sasvim jednostavna svaka jedinica se pretvara u nulu a svaka nula u jedinicu NOT 11010101 je 00101010 Jedna od primana negacije je promena znaka naučili smo de se *prvi komplement* broja dobija kaoe se svaka jedinica u tom broju pretvori u nulu a svaka nula u jedinicu, sada znamo de se oveko pretvaranje naziva *negacijem broja*. Ukoliko nam je potreban *drugi* ili *potpuni komplement* broja, rezultatu negacije ćeme prosto dodati jeden

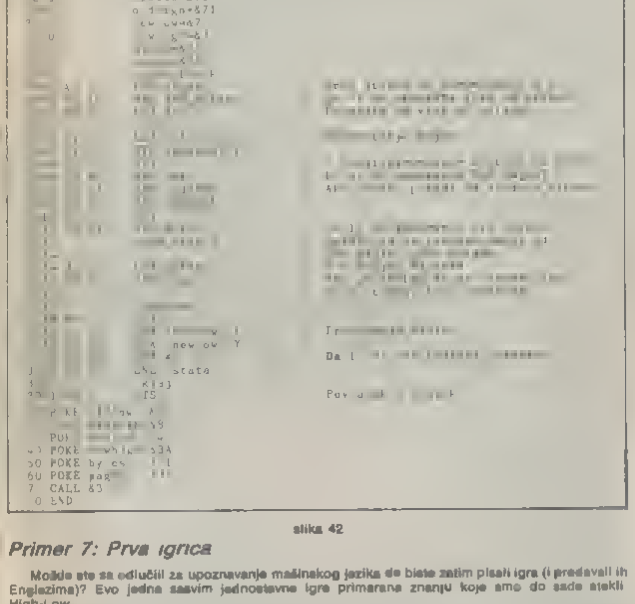
Šiftovanje

Posmotrajmo brajeva 115, 230 i 204 i pokušajmo da uočimo neku vezu izmedju njih Na prvi drugi i treći pogled veze nema. Ukoliko imate malo smisle za enigmatiku sejćete se de govorimo o logičkim operacijame sa binarnim brojevime pe ćete naše brojeve predstaviti u tom obliku i dobiti %01110011 %11100110 i %11001100 Vzas postaje očigledne drugi broj je dobijen kade smo odbacili prvu cifru prvog i dodali mu nulu na kraj treći broj je dobijan kada amo isto to uradili sa drugim Sledeći broj u nizu bi dekle bio %10011000 ili 152

Tvrdnju *sledeći broj se dobija kada prethednom dopišemo nulu na kraj i odbacimo njegovu prvu cifru* možemo da iskažeme i na drugi način *sledeći broj se dobija kada sve cifre prethednog*

Na slici 42 vidimo program koji premešta blok memorije proizvoljna veličina pridodali smo bejzik program koji izazive pranos sadržaje ćelije &8000 — &9021 u ćelije &3A40 &3A41 . Svaka je adresa podeljene na dve bajta viši i niži Tako imeme promenljive *oldlow* i *oldhigh* (niži i viši bajt adrese početka bloka), *newlow* i *newhigh* (niži i viši bajt adrese na koju blok treba de se pramsati) i na kraju *pages* i *bytes* — promenijive kaje određuje koliko se bajtova memorije premešte (u našem pdmeru &1021) Prve četiri promenijive merađu da budu smeštena na nultu stranu sećamo se de indirektnu adresu mora de čuva neki od prvih 256 bajta memorije

Program će raditi i u slučajevima kaoe se zahteva pranošenje manje od 256 bajta pe čak i ake zahtevamo pranos 0 bajte Program međutim neće uvek raditi eko sa blokovi preklapaju premeštanje ćelije &2000—&3000 u ćelije &2100 — &3100 će steti neželjene rezultate Pokušajte de medfikujete program teko de korektne reši i ovaj problem!



slika 42

Primer 7: Prva igrica

Možde ste se odlučili za upoznavanje mašinskog jezika de biste zatim pisali igre (i predavali ih Englezima)? Evo jedna zasavim jednostavne igre primarana znanju koje ame do sada stekli High-Low

Zamislite broj između 0 i 1023 i startujte nas program Računar će ispisati neki broj e zatim ćete primanuti V eko je broj koji ste zamislili veći M ako je manji e P ako je računar pogodio Videćete da će svaki broj koji zamislite biti pogođen u najviše 10 pokušaje

Igra je zasvim nezavisna od bejzika što znači da ulaz i izlaz obavlje sopstvenim potprogramima koji se svoje strane pozivaju rutina iz ROM-a čije adrese možete da saznate konsultovanjem knjiga tipe „ROM Disassembly „Advanced User Guide ili naših ranijih umataka Potrebni su vam potprogrami koji ispisuju ASCII znak ne ekranu učitavaju slovo sa tastature i ispisuju sadržaj dve memorijska ćelije ne ekranu kao dekadni broj nešto se slično sasvim sigurno nalazi i u ROM-u vašeg kempjutera

Šio se algoritme ne kome pošive program High Low tiče nećemo mu obraćali mnogo pažnja radi se o obidnom binarnom pretraživanju Ukaliko vam je termin binarno pretraživanje stran pogledajte meli bejzik pregram koji je potpuno akvivalanian bitnom delu našeg mašinu

```
3000            ascii_hex
3000 20 15 30 JSR konverzija_cifre      .  Nizi bajt je u akumulatoru -
3003                                    .  treba ga prevasti u binarni
3003                                    .  broj
3003 8D 1F 30 STA lswnibble             .     i upisati ga u 'lswnibble'
3006 8A         TXA                     .  Visi bajt je u X -
3007 20 15 30 JSR konverzija_cifre      .  - prevaden u binarni broj
300A 0A         ASL A                   .     i ua ujega je dapisae
300B 0A         ASL A
300C 0A         ASL A
300D 0A         ASL A
300E 0D 1F 30 ORA lswnibble             .     sadrzaj 'lswnibble'
3011 8D 20 30 STA fiual                 .  Rezultat u memarlju
3014 60         RTS                     .  Povratak u bejzik
3015
3015            kauverzija_cifre        .  Petpragree kaji kauvertuje
3015                                    .  ASCII ausk ia akueulatara u
3015                                    .  binardnu vreduost
3015 38         SEC
3016 E9 30      SBC #$30                .  Oduzies ae ASCII "0
3018 C9 0A      CMP #$A                 .  Ake je rezultat veci od 10
301A                                    .  radile se o cifri A   F
301A B0 02      BCC kraj
301C E9 07      SBC #7                  .     pa treba oduzeti
301E                                    .  ASCII "A" - ASCII "9" + 1
301E            kraj
301E 60         RTS                     .  Povratak u glavni program
301F 00         lswnibble EQU$ 0
3020 00         fiual     EQU$ 0                                      4
```

slika 40

Primer 5: HEX-ASCII

Kao što nam je potrebno da pretvorimo niz znakova u broj, tako će nam zatrebati da pretvorimo broj (upisan u akumulator) u dva ASCII znaka koje će, ispisana na ekranu predstavljati heksadekadnu vrednost. Ovu konverziju možete da povarite programu HEX ASCII koji je dat na slici 41. Program tepo ilustruje tehniku koju zovemo maskiranje. Izdvajamo četiri niža bita, pravodimo ih u znak, a zatim šiftovanjem izdvajeme signifikantnije četiri bita i pretvaramo ih u drugu (zapravo prvu) cifru

```
3000            hex_ascii
3000 48         PHA                     ;  Akumulatar se cuva ua steku
3001 29 0F      AND #$0F                ;  Izdvaja se nizi uibi akumulatore
3003 20 15 30 JSR kouverzija_uiblu      ;     i konvertuje u ASCII znak
3006 8D 1F 30 STA niza_cifra            ;     koji se smeata u memarlju
3009 68         PLA                     ;  Izdvaja se visi uibi akumulatora
300A 4A         LSR A
300B 4A         LSR A
300C 4A         LSR A
300D 4A         LSR A
300E 20 15 30 JSR konverz ja_ulbia      ;     i konvertuje u ASCII znak
3011 8D 20 30 STA visa_cifra            ;     koji se smesta u memarlju
3014 60         RTS                     ;  Povratak u bejzik
3015
3015            konverzija_nibla        ;  Petpragram kaji konvertuje nibl
3015                                    ;  upisan u uiza 4 bita akueulatara
3015                                    ;  i vreca njihavu ASCII predatavu
3015 18         CLC
3016 69 30      ADC #$3C  0
3018 C9 3A      CMP #$3C  9 +1          ;  Da li je radi o cifri A F?
301A 90 02      BCC okey
301C 69 06      ADC #6                  ;  Ako da   treba dodutl 7 (6+carry)
301F            okey
301E 60         RTS                     ;  Pavratuk u gluvui pragres
301F 00         niza_cifra EQU$ 0
3020 00         visa_cifra EQU$ 0
```

slika 41

Primer 6 Premeštanje bloke

Nezgodne strane 6502 je što nema čeenaestobitnih registara — za bilo koju operaciju nad blokom memorije koji je veći od 256 bajta moramo da se mučimo se postindeksiranim indirektnim adresiranjem. Ovo je adresiranje na prvi pogled veoma složeno morate da praiite promenu višeg i nižeg bajta adrese i još de se brineje o Y registru. Savetujemo vam ipak da se što pre privikhete ne indekairana adresiranja bez njih se jednostavne ne može pregramirati na &6502

pomerimo ulevo za jedno mesto a na upražnjene mesto poslednje cifre upišemo nulu. Ove je ujedno i najjednostavnija definicija šiftovanja u levo

Šiftovanje udesno je vrlo slične operacije pri čemu bi se od broja %01110011 dobio najpre broj %00111001 a zatim broj %00011100 Ukoliko se radi e pozitivnim brojevima šiftovanja ulevo predstavlja množenje se 2 e šiftovanje udesno — celobrojno deljenje se 2

Procesor 6502 omogućava relativno složene logička šiftovanja brojeve o kojima ćemo još dosta govoriti Za sada je važno semo znad da se kod tih šiftovanja vedeći III krajnji bit ne gubi već se pranosi u ledan od flegova nazvan *Carry* Postoji mogućnost i da se bit iz Carry flega vrad u šiftovani broj e takođe i da se izgubljeni bit vrati ne kraj broja tako bieme od %10010001 dobili%00100011 Ovakva operacije se iz očigledniih razloga naziva rotiranjem nalevo

Instrukcije i asembleri

Kada biste program koji emo napisali u prethodnom poglaviju pokušali da „saopštite mikroprocesoru koji upoznajemo razultati bi bili nikakvi Pra svega, za zadavanje instrukcije smo koristili neke reči LOAD STORE COMPARE, BREAK ali emo prečutkivali način na koje bi one mogle da se unesu u računar Možda da ih otkucate umesto bejzik programa? Pokušajte i pozdravićе vas 'Syntax error' III pošto ste nečuli da se mašinski programi unose primenom instrukcije POKE da otkucate POKE LOAD? Sličan pozdrav! Da bi računar mogao da izvrši mašinski program on mu mora biti seopštan u prihvatljivom kodiranom obliku

Instrukcije se kodiraju u binarne brojeve pri čemu kod aveka instrukcije obično zauzime jedan bajt. Deim koda koji bi ne primer, zamenjivae skraćenicu LOAD računaru trabe da saopštimo i adresni deo naše instrukcije (sećate se onoga LOAD A, #2567>, se koji se obično koriste sledeći, aukcesivni bajtovi memorije. Različite instrukcije, sejedno se svojim adresnim delom, mogu da zauzmu jedan dva lli više bajtova pri čemu ovaj broj varira od instrukcije do instrukcije

Kada se već instrukcije moraju pratvarati u brojeve zašto izmišljati reči koje te brojeve zamenjuju? Zato šte je mašineki jezik I onako po prirodi težak I nečitljiv pa postojanje naziva instrukcija (tzv *mnemoničkih skraćenica)* olakšava kako pisanja programa tako i njegove docnija prapravljenja Samu transformeciju reči u binarne brojeve je u stanju da obavi kompjuter pomoću programe koji nazivemo *asembler* i kome posvećujemo kraj ovog poglavlja. Mnemoničke skraćenice su stvar konvencije možemo da se odlučimo da instrukciju čiji je kod &AP zovemo instrukcija I asembleri

LDA III LOAD lii samo L i bilo koja od tih varijand će rezuidrati istim mašinskim programima. Slično je i u bejzieu naredba PRINT bi sasvim lapo mogla da se zove WRITE DISPLAY III TYPE. Konvancije su medutim jednom usvojene I ne možemo de ih menjamo pe se ne njih treba prtvidi. Posle određenog broja seći sedenja uz računar, neućšete sve bitne skraćenice naparne! I kucati ih bez mnogo razmišljanja.

Korišćenje mnemoničkih skraćenica krije u sebi veliku opasnost tendenciju da se uobrazi postojanja nekih neredbi Ukoliko ne primer, znate da postoji naredba JUMP (&2000) moguće je da po analogiji upotrebite I naredbu CALL (&2000) koja, na primer ne postoji! Program koji eme nazvali asemblerom će vas jasno upozoriti na greške ali vam to neće mnogo pomoći ako ste čitav segmenl programe zasnovali na nepostojećoj naredbi Osmobitni mikroprocesori zbog uštede memorije i jednostavnosti konstrukcije imaju razne modalitete adresiranja koji au pristupačni samo kod nekih instrukcija I za njihovo korišćenje je potrebno, bar u početku neprestano konsultovati tabelu naredbi Korišćenje te tabele ćete upoznavati kako ovaj tekst bude napredovao, za sada samo bacite jedan pogled ne nju I ukoliko za to imate uslova napravite par fotokopija — te će fotokopija postati redovan gost na vašem pisaćem stoiu U tabeli pobrojene skraćenice instrukcija njihovi kodovi i načini adresiranje koje ih prate dali smo najzad I kratak opis delovanja svake naredbe

Memorije i adresiranje

Memoriju možete da zamislite kao niz ćelija se slika d Svaka ćelija ima svoju adresu koja se kreće izmedu 0 I 65535 (lli ako više volite heksadekadne brojeve, izmedu &0 I &FFFF) Svaka ćelija se sastoji od osam bitava koji su obeleženi brojevima 0—7 Adresni prostor od 65536 bajta ili 64 kilobajta mora da bude podaljen na ROM u koji će biti smešten operativni sistem računera i interpreter(i) programskih jezika kao što je bejzik I RAM u koji će biti upisivani programi koje korisnik piše Za sada ćemo reći da se ROM kod računara se 6502 obično počinje od adrese &8000 ili &C000.

Većine instrukcija koje zadajemo mikroprocesoru se bavi manipulacijom (pranošenjem sabiranjem restiranjem) brojeva koji su smešteni u neke memorijske ćelije što znači da se uz kod instrukcije mora navesti I neka oznaka ne osnovu koje će računar zakijučiti gde se nalaze podaci koje trabe da određuje Te oznake se zove *adresni deo instrukcije* U zavisnosti od načina na koji mikroprocesor transformiše adresni deo instrukcije u podatak se kojim treba manipulisati razlikujemo razne *modove adresiranja* od kojih ćemo neka upravo upoznati Vrlo je verovatno da slejeće redove nećete odmah razumeti previše novih pojmova za početnika Ne ipak prelatite lii pogledom I zapamtite da ćete posleje vratiti se na adresne modove kada budete upoznali sve instrukcije

slika 6

Najjednostavniji slučaj adresiranja je izostavljanja adresa — adresa je tada implicirana smislom naredbe pa se ovakvo adresiranje naziva *impliciranim (implied)*. Posmatrajmo na primer naredbu SEC čiji je jedini smisao da setuje (postavi na jedinicu) ‚carry' flag koji smo već pomnjali iz koda ova naredbe se zna sve što je potrebno za njane izvršavanja, pa bi bilo kakva adresa (npr $EC carry) bila suvišna

Slično implicitanom je *akumulatorsko adresiranje* koje podrazumeva da se argument nalazi u akumulatoru, jednom od registara mikroprocesora Instrukcija ROL A na primar rotira sadržaj akumulatora ulevo

Malo je složeniji slučaj instrukcija LDA #25 koja u akumulator dovodi broj 25. Broj kojim trabe manipulisati se ovde nalazi u okviru sama Instrukcije, pa se takvo adresiranja naziva *neposrednim (immediate)* i obeležava tarabom (#) instrukcije ovoga tipa ćemo koristiti kada želimo da dovodimo konstanta u memoriju smeštamo ih u registre ili na neki drugi način manipulišemo njima

Došlo je najzad vreme da počnamo da redimo se sadržajima mamorijskih ćelije I upoznamo *apsolutno adresiranje* posmatraćemo Instrukciju LDA &1000 koja u akumulator dovodi osam bita ćelija čija je adresa &1000

Procesor 6502 poznaje još jednu vretu apsolutnog adresiranja *apsolutno adresiranja nulte strane* O čemu se radi? Konetruktori mikroprocesora koji upravo upoznajamo su odlučili da odvoja 256 bajta memorijska mape za ‚pamćenja podataka koji su veoma potrebni pri radu kompjutera tim se podacima prilazi brzo I uz minimalan utrošak mamorije Sasvim je logično da se za privilegovene bejova proglase memorijska ćelije čije su adrese izmađu 0 i 255 (0 i &FF) — dok instrukcija LDA &1000 zauzima tri bejta LDA &10 zauzima svega dva! Za sada nama mnogo potrebe da razmišljate o ovoj konvenciji- pristupajte ćelijama koje su vam potrebne e asembler će se pobrinuti za maksimalne moguće uštede u prestoru

Ako postoji *apsolutno adresiranja* verovatno postoji I *relativno!* Uprave tako kod apsolutnog edresiranja specificiramo memorijska ćelija nevodeći njihovo tačno mesto u memorijskoj mapi dok kod relativnog navodimo restojanja podataka od matrukcija koja se izvršava. Prototip ovakve Instrukcije bi bilo BEO PC+&20 PC je skraćenica od *Program Counter* to je ima registra koji prati izvršavanja programa I stalno sadrži adresu Instrukcija koja će se sledeće izvršavati TA adresa će biti sebrana se konstantom &20 a zatim će pod određanim uslovima biti izvršan skok na naredbu čiju emo ovim sabiranjem dobili Za šta bi mogla da se iskoristi ovakve komplikacija? Ukoliko napišemo čitav program koristeći ovakvo edresirenja za sva naredbe (posebno naredbe skoka) on će se koraktno izvršavati bez obzira ne to u koji ga segmant mamorija upišemo Kako ovakve mogućnost nija naročito korisna relativno adresiranja je kod jednostavnih procesora kao što je 6502 zastupljano samo kod malog broja instrukcija

Nezgodna strana svih edresiranja koje smo do sada upoznair ja što edresa moreju da se znaju u dobe pisanja programa Varujući da je većina čitalaca ovog umatka dobro upoznata se bejzikom objesnićemo problem na primanu naredbi PRINT A(100) i PRINT A(i) Obe naredbe treba da prikažu ne akranu sadržaj nakog elamanta niza A Kod prve (PRINT A(100)) ja broj tog alamanta fiksiran u okviru naredbe pa ona predstavlja primar *apsolutnog* adresiranja Naredbe PRINT A(i) se druge strana nareduje bejzik interpreteru da najpre proveri vrednost promanljive a anđe da lapiše vrednost alamanta niza čiji broj ona čuva To je *indirektno* adresirenje koja se u šarmineloglji mikroprocesora 6502 zove *indeksiranjem*

Pogledajmo Instrukciju LDA &2000 X koja zahteva dovođenja sadržaja nake mamorijske ćelija iz akumulatjor U okviru seme te instruk lje nalazimo tav *bazni adresu* (&2000) dok registar X sadrži *ofset*. Računar sabira bazmu adresu i ofset a zatim priatrpa ćelji čija je adresa odradana

Program će korektno rediti I ako ja množilac nula — proći će kroz seriju šiftova i vratiti kontrolu bejziku

Na sljci 39 je i jednostavan bejzik program koji komunicira se mašincem primetite način na koji se, funkcijama MOD I DIV izdvajaju niži i viši bajt crnlioca r kako se docnije četiri bajta rezultata pretvaraju u dekadnu vrednost

```
3000 A2 0B      LDX #11
3002 BD 3  6    LDA ,X
3005 9D 3A 3B   STA ,X
3008 A2 10      LDX #1
300A            slic1
300A 4E 36 30   LSR
3000 6E 35 30   ROR
3010 90 10      BCC
3012 AD 39 30   LDA
3015 18         CLC
3016 6D 33 30   ADC
3019 8D 39 30   STA
301C AB 3A 30   LDA ,Y
301F 6B 34 30   ADC
3022            rotiraj
3022 6A         ROR A
3023 8D 3A 30   STA pred4
3026 6E 39 30   ROR pred3
3029 6E 38 30   ROR pred2
302C 6E 37 30   ROR pred1
302F CA         DEX
3030 00 DB      BNF sledeći
3032
3032 60         RTS
3033
3033 00         mnozenik_low  EQUB
3034 00         mnozenik_high EQUB
3035
3035 00         .mnozilac_low  EQUB 0
3036 00         mnozilac_high EQUB 0
3037
3037 00         pred1         EQUB 0
3038 00         pred2         EQUB 0
3039 00         pred3         EQUB 0
303A 00         pred4         EQUB 0

 10 INPUT "Mnozenik  " A
 20 INPUT "Mnozilac  " B
 30 POKE mnozenik_high,A DIV 256
 40 POKE mnozenik_low, A MOD 256
 50 POKE mnozilac_high,B DIV 256
 60 POKE mnozilac_low, B MOD 256
 70 CALL mnozilk
 80 PRINT "A*B (BASIC)   = ";A*B
 90 PRINT "A*B (MASINAC) = "
100 PRINT PEEK(pred1)+256*PEEK(pred2)+256 *PEEK(pred3)+256^3*PEEK(pred4)
110 PRINT
120 GOTO 10
```

slika 39

Primer 4: ASCII-HEX

Kada poželimo da pišemo mašinske programa koji se neće oslanjati na ulazne I izazne naredbe bejzika zatrebeće nam mnogo različitih konverzija Često je na primar potrebno pretvoritr dva ASCII znaka koji predstavljaju heksadekadni broj u vrednost pogodnu za računanja Jedan od jednostavnijih načina da se ovaj problam reši pokazuja I naša slika 40

Pre pozivanja potprograma ASCII-HEX treba u X registar upisati višu, a u Y registar nižu cifru broja koji se konvertuje — cifre se jsano upisuju kao ASCII znaci 0-F Program će konvertovati ove dve cifre r vratiti razultat u akumulatoru Da bi se očuvala jednostavnost, nija kontrolisano prekoračenja ukoliko zahtevate od računere da konvartuja niz znakova GG u binarni broj dobićete nepredvidljiv rezultat

Vredi primaiti na koji su način spojene dve nibla rezultata signifikantniji je šiftovan za četiri mesta ulevo a ehos mu je jednostavnim ORA prikrpljan niži nibl Primanili ame dakle jahniku kaja je na naki način obrnuja maskiranju

Dokažja da sia razumali ovaj program pišući odgovarajuću rutinu koja u akumulatar davodi vrađnosi nize od pr! A@Cli znaka (iz A X i Y) koji su shvađeni kad eso dakadni broji

postoji. Ovakav interapt koriste autori igara de bi omogućili simultano odvijanje nekoliko redniji — programi koji upravljaju kretanjem raznih mimeiruma veoma često predstavljaju deo interapt rutina dok sa pozitivec koje vi kontrolišete kreće uz pomoć glavnog programa. Ne interapjime se zbenive i mogućnost nekih računara da lapisuju iakst na štampaču dok se korisnik zabavlje pertijom šaha ili de svireju dok kucate program. Sve u svamu inrerapti su izuzetno privlačni i korisna stver (na semo se hardverskoj) i vredi ih proučiti deleko podrobnije

Biblioteka programa

Primer 1. Bilo šte

Počećemo od sasvim jednostavnog primera. želimo de računar izvrši sasvim kratak mašinski program koji redi bilo šta u našem slučaju komplementre (zamenjuje sveku nulu jedinicom a jedinicu nulom) broj upisan u memorijsku ćeliju čija je adresa &3100. Sa slika vidimo da bajzik program sa tastature uzima neki broj a onda ga primenom naredba POKE upiše u ćeliju &3100. Zatim sa primanom naredbe USR poziva mašinski program koji je smešten počevši od adrese &3000 (možete ga jasno smestiti i na mnoga druga mesta) i na kraju lapisuje nova vrednosi bajta &3100. isprobajte ovaj program — pomoći će vam da bolje upoznete vaš asembler i njegova vese se bajzikom

```
3000 AD 00 31 LDA &3100        10 tNPUT A
3003 49 FF    EOR #&FF         20 POKE &3100,A
3005 8D 00 3t STA &3100        30 CALL &3000
3008 60       RTS              40 PRINT PEEK(&3t00)
                               50 GOTO 10
```

slika 37

Primer 2. Vremenske petlje

Primer 9 predstavlja vremeneku petlju — ako vam se ponekad učini da vaš mašineki program redi prebrzo usporite ga ovakvim dodatkom. Trajanje vremenske petlje sevisi od počatne vrednosti u akumulatoru — ukoliko u njega upišete naki valiki broj (ne primer &FF) petlje će potrejati pedesetak sekundi (ako računar redi ne kloku od 2 MHz). Upisivanje jedinice u akumulator će se druge strane izazvati jedva primetno čekanje.

Zanimljiva je činjenica de amo u X i Y registar upisivali nulu a zatim koristili DEX (ili DEY) i BNE ciklus. Posle prvog umanjivanja nule postaje &FF a onda se sadržaj registra smanjuje sve dok ponevo ne stigne do nule. Verovetno bi stvar bila razumljivija da amo nepisali LDX #FF ali bi tada petlje trajale za mrvicu kraće!

Ovaj primer treba da nam dočare i brzinu mašince. isprobajte i odgovarajuću bajzik rutinu i uporedite vremena izvršavanjal

```
3000 A9 FF    LDA #&FF        . Nuuji broj deje duzu petlju
3002          ciklust
3002 A2 00    LDX #0
3004          ciklus2
3004 A0 00    LDY #0
3006          ciklus3
3006 88       DEY
3007 D0 FD    BNE ciklus3
3009 CA       DEX
300A D0 F8    BNE ciklus2
300C 18       CLC             Oduzimanje 1 od akumulatora i
300D E9 00    SBC #0            A = A - 0 - (1-C) = A - 1
300F D0 Ft    BNE ciktusi
30ti 60       RTS
```

slika 38

Primer 3· Šesnaestobitno množenje

Obećali amo vem „previ program se množenje množe se dva šesnaestobitna broja i dobije jedan irideseidvobitni

Program je relativno kratak pa treba utrošiti po neku reč na njegovo detaljnije obrazloženje. Potrebne su nam pre svega po dva memorijske ćelije za množanik i množilac i čak četiri ćelije za rezultat. Kao i ranije izdvajamo bli po bit množioce ignorišući nule. Kada pronademo jedinicu dodaćemo množenik proizvodi (oba bajta) a zatim čiftovati proizvod sa jedan bit udesno ovim zbirom. Ako se na primer, u registru X nalazio broj &33. naredba LDA &2000,X će u akumulator dosesi sadržaj ćelije čija je adresa &2000+&33=&2033. Nevolje je što se u registar X može upisati samo broj između 0 i 255 tako da u datom primeru promenom vrednosti koju upisujemo u X možemo da pristupamo samo ćelijama od &2000 do &20FF. Za operacije se većim blokovima memorije treba de upoznemo *post indeksirano indirektno adresiranje*.

Primer *post indeksirenog indirektnog adresiranja* je naredba LDA (&20),Y. Mikroprocesor će izvršavajući ovu naredbu najpre prisiupiti memorijskim ćelijama &20 i &21 naka je u prvoj našao broj &55 a u drugoj &70. Pcametrane dva broja spajamo i tako dobijamo &7055. Mikroprocesor će sada na ovaj broj dodati sadržaj Y registre (eko se u Y nalazio broj 9 rezultat je &7058) a onda prisiupiti tako ćeiljl čije je adresa ne taj način dobijena. Operacije je sva u svemu vrlo kompiikovane ali ćemo se ne raju postepeno navikavati jer je ze programiranje neophodne.

Procesor 6502 pozneje još par modova adresiranje (npr *preindeksirano indirektno*) ali se oni retko koriste pa ih za sada nećemo pominjati

Asembleri

Ulogu asemblera (ili asembler programa) već poznajemo. oni trebe de prevedu program pisan mnemoničkim skraćenicama (zvaćemo ga *izvorni program*) u *objektni kod* — pravi mašinski program koji će biti upisan negde u memoriju. Ukoliko nemerevate da vam ovaj umetak posluži se bilo šta ceim za ukras *nabavite program za vaš kompjuter*. To nije naročito valika investicije (ovakvi programi su kod preprodaveca jeftini valjda zato što slabo koga interesuju) a predstavlje *neophodan* uslov ze rad ne mašinskom jeziku. Pretpostavljući da asembler već posedujete nećemo se baviti prevođenjem instrukcija na pravi mašineki jezik. ukoijko dođete do toga da vem nešto takvo setreba verovatno ćete biti dovoljno dobar poznavalac mikroprocesora da se semi snađete. Ostatek ovog poglavlje je seto posvećen iskljućivo upotrebi asemblera.

Asembler je program kao i svaki drugi. pravi se u mnoge verzija i svaka od njih treba da ima svoje karakteristike i svoje uputstvo za upotrebu. Računari su se bavili asemblerima za komodor 64 ukolika koristite neki drugi računar ili asembler o kome nismo govorili morećete da pročitate originalno uputstvo što vam, u svakom slučaju sevetujemo. Videćete da se asembler sastoji iz *editora* koji omogućava da unosite ispravjate i snimate program na traku ili disketu i *prevodioca* koji ovako formiran tekst (kažemo tekst, jer su mnemoničke skraćenice obične reči) prevodi u objektni kod i upisuje ga u memoriju. Editori se razlikuju od asemblera dok prevodioci rade manje-više jednako što znači da ćemo im posvetiti posebnu pažnju.

Da bi prevodilac mogao da smesti program u memoriju treba mu saopštiti adresu od koje program počinje. Ta adresa se naziva *origin* i obeležava se ORG ill po konvenciieme firme MOS Technology tačkom. Ukoliko se na početku programa pisanom mnemoničkim skraćenicama nađe ORG &3000 (ili ×3000) rezuitujući mašinski kod će biti upisanu memoriju počevši od adrese &3000.

Reč ORG nije mašinska instrukcija pošto se ona na prevodi u binarni broj koji bi bio smešten u memoriju. Ona umesto ioga daje prevodiocu uputstvo o tome šta da radi se pravim instrukcijama pa se nazive *asemblerskim direktivom*. Asemblerskih direktiva jasno ima mnogo pa ćemo ih upoznevati kako nam budu trebale. Neka od njih imaju sasvim banalno i se sepakta posle koji treba de se otkavi nekitno značenje de ili će se pri asemblerinju listing prikazivati na ekranu ili će se ispisivati na štampaču da ii će računar prijevljivati eve greške ili semo one fatalne i slično. Neke druge direktive (kao što je ORG koju amo upravo upoznali) su fundamentalno važne.

Osim asemblerskih direktive i mnemoničkih skraćenica. u asemblerskim listinzima mogu de se nađu i *pseudo natrukcije*. Pseudo instrukcije leko na predstavljaju ni jednu od naredbi koju bi mikroprocesor razumeo u rezuitujući kod upisuju određeni sadržaj. Pretpostavimo ne primer da naš program treba da ispiše tekst „ZDRAVO" na ekranu. Isplavljanje se uzećemo najjednostavnijl slučaj. obavlje tako što se poruka ZDRAVO slovo po slovo prepisuje u video memoriju što znači da slova Z, D R A V I O moreju da budu upisana negde u RAM. Program koji rešava problem bi mogao de izgleda otprilike kao na slici 7

```
        LDX #0          registar X će brojeti ispisane znakove
petlje
        LDA poruka X    sledeća slova poruke u akumulator
        BEQ kraj        ako je u akumulatoru 0, ide se na dea
                        programe ozačen se KRAJ
        JSR OSWRCH      slova se ispisuje na ekranu
        INX             X se povećava za 1
        JMP petlja      ispisuje se sledeće slovo
kraj
        RTS             kraj rada
poruka
        EQUS "Zdravo"   poruku koju treba ispisati
        [illegible]     ka[illegible] na početak reda
        [illegible]     ka[illegible] i sledeći red
        [illegible]       [illegible] kraj poruke
```

slika

Nije previše važno da shvatite kako tačno radi ovaj program, u globalu posmatrano računer šalje slovo po slovo poruke i ispisuje ga na ekranu. Obzirom da bi ispisivanje bilo mrtva petlja ne kraj poruke je upisan nula bajt koga program detektuje (naredbe COMPARE CMP) i prekida sa radom

Obratimo pažnju na iznenadne pseudo naredbe EQUS i EQUB. Prva od njih označava da prevodilac treba u memoriju da upiše takst (string odatle ono S) ZDRAVO a svako slovo u po jednu memorijsku ćeliju koristeći naravno ASCII kod. Slično tome pseudo naredba EQUB zahteva od prevodioca da u memoriju upiše bajt 00. Ukoliko nekom prijatelju docnije budete dali mašinski program poput ovoga on će perfektno raditi. Međutim, eko taj prijatelj bude pokušao da shvati kako ja program napisan tj. da ga pratvori u mnemoničke skraćenice naći će se u valtkom čudu kada buda pokušao da praveda (odnosno kako se to stručno kaže *disasemblira*) deo programa koji predstavlja poruku ZDRAVO. Naći će, naime na bajtova od kojih neki predstavljaju instrukcije dok su drugi nedefinisani. Čak i oni koji predstavljaju instrukcije će imati besmislene adresne delova i uopšte neće ličiti na program. U slično situaciji bi se nalazio i sam mikroprocesor da amo zaboraviti na naredbu RTS koja prekide njegov rad pokušao bi da izvrši tast ZDRAVO kao mašinski program i pri tom verovatno upao u neku vrstu beskonačne petlje

Iz čitava ova priča može da se izvede jeden izuzetno važan zaključak od izvornog teksta programe se jednoznačno dobija objektni kod uz pomoć programa koji nazivamo asembler. Obrnuti proces dobijenje mnemonički pisanog programe iz objektnog koda (disasemblranje) nije jednoznačan i osim kod asavim jednostavnih programe nije ni malo lak. Vidimo na primer da amo u programu koji smo pisali oznešili izv *labeleme* (o njima docnije) tri meeta koje amo nazvali petlja kraj i poruka pri čemu ti nazivi otpadka asocinaju na nemenu ozneka. Ukoliko neko disesamblira ovaj program nače nikako moći da zna za ova nadre ozneke pa će mu pređenje čitava atvari bit daleko komptikovanije. Zato uvek i obavezno čuvajte ne tragi iti diskati izvorna varzija svih programa koja napišete ta verzija će vam biti dragocena eko jednom poželite da promenite nak avot program iti da ga uktoptta u nesto drugo

U asemblarskim listnizme se kao što amo videli pojavljuju i komentari koji su od korisnog dale naredbe obično odvojeni simbolom. Po pravilu se komentariše aveka instrukcije što ne prvi pogled izgleda prilično dosadno. Pokazalo se međutim da kucanje komentera može de pomogne na samo u docnijim modifikacijama programe nago i u otkrivanju grešaka eko sažmete funkciju neke naredbe u jednu rečenicu možda ćete primatiti neku nepravilnost u njoj! Bajzik program koji ptšate se komantarima ja aporiji i zauzima daleko više memor je od bajzik programa bez REM-ove rasleg je u toma što se avako slovo komentera upisuje u memoriju baš kao i sveko slovo instrukcije. Za razliku od toga asembler u objektni kod prevodi samo korisne delove instrukcija što znači da komenteri bivaju potpuno ignorisani. Program koji dobro opširno komantarišete neće, dakla biti ništa duži ni sporiji od nekomentarisanog programe biće jedino mnogo razumljiviji što i nije tako mala stvar!

Osim asemblara za učenja mašinskog jezika će vem dobro doći program koji se naziva *monitor* ili *debager*. Ovaj program vam omogućave de pretražujete memoriju i menjete njen sadržaj i što je posebno važno izvršavate mešinske programe instrukciju po instrukciju posmatrajući promene u registrime i memoriji. Iakuen programeri koriste ovakve programa za traženja grešaka i tako štede svoja vreme dok će za početnike svekako biti interesantne mogućnost da u nekoj vrsti usporenog filma proučavaju delovanje naredbi koje upoznaju. Reći ćemo ipak, da upotrebe monitor programa obično nije sasvim jednostavne i da se njime treba de se upoznate tek kada ber pojmovno ahvatita osnove mašinskog programiranje

Pra nago što predata na čitanja sledećih poglavlja koja će se baviti naredbama mikroprocesora 6502 trebe de rešita nekoliko praktičnih problema koji će proistaći iz činjenice de programi koje ćemo sastavljati mogu da se izvršavaju na računeru što znači de je pošeljno da neke od njih i otkucate. Pra toga morate da proverite ne koje memorijska adrese treba de smeštate mešinske programe keko sa oni na tr , kosili se asemblerom i izvornim kodom (eko se na primer asembler nalazi u memorijskim ćelijama čija su adresa &7000—&8000 direktiva ORG &7500 će skoro sigurno izazvati krah čitavog sistema jer će sasmbler asembtirajući program uništiti semoga sebe). Treba osim toga da proverite kako se mašinski programt pozivaju iz bejzika (za to obično služi naredba CALL iti funkcija USR) i kako se iz njih vrače u bejzik (obično mešinskom naredbom RTS)

Arhitektura 6502

U uvodnom emo poglaviju objasnili pojmove *registar* i *fleg* došlo je vreme de te pojmova upoznamo i ne konkratnom primaru — arhitekturi mikroprocesora 6502

Procesor 6502 ja opremljan sa tri registra opšta i isto tolliko registare specijalna namene ovaj broj kada pe uporedimo sa Z80 i nakim drugim osmobitnim mikroprocesorime ne izaziva preveliko poštovanje— ali na budtmo zabrinuti — videćemo de je 6502 koncipiran teko de mu mnogo registara nije naročito ni potrabno. Dok programer koji počinje de radi sa Z80 može sebi priuštiti lukauz de privrameno zaboravi neka registre koristeći 6502 moraju de utože meio trude i dobro zapamte šta su im konstruktori tirme MOS Technology stavili na raspolaganje

Upoznati svrhu registare opšte namene nije teško u njih smeštamo podetka sa kojima operišemo. Jedan od tih registara (A) amo već upoznali u njemu se obavlja većina aritmetičkih i logičkih operacija. Ostali registri opšta namene su takozvani indeks registri koji se obeležavaju

Beakarvliskog pas?) usred noći mehao svebom na prozoru e robijaš Selden iz udetjene močvara posmatrao signel da bi poslao odgovor radilo se o „hend shekingu" mikroprocesor u neku fiktivnu memorijsku lokaciju upisuja podetak i čeka de periferijski uređaj taj podetak pokupi. Kada periferija pokupi podetak mikroprocesoru se šalje signal preko nakog drugog porta i on nastavlje se normelnim redom. Ove je tehnika daleko od savršenstve u šta ćemo se uvariti na jednostavnom primeru se slike 36

De vidimo kako ovaj program funkcionše. Protokolom je predviđeno de printer kada je spremen za prijem sledećeg znaka resetuje sedmi bit onoga što računer emetra memorijskom ćeljom &FEt1, e de tade računer slovo koje trebe štampeti upiše u ćeliju &FE10. Program periferijen učitave vrednost sa porta 255 i testira njen sadmt bit dok je on setovan eva se okriće u mrtvo petlji. U trenutku kade štampač bude spraman mikroprocesor će mu postati sledeće slovo i ponovo čekati de se zahtevene operacije izvrši

Ovakvo rešenje je primenjivo eli sporo — mikroprocesor se vrti u beskonačnoj petlji čekajući da štampač završi svoj posao i za to vreme ne može de radi ništa drugo. Bilo bi daleko bolje kade bi mikroprocesor zedeo posao štampaču a zatim radio našto drugo dok štampač ispisuje slovo. Printer bi po završetku posla nekako dojevio mikroprocesoru de je ponovo spremen i ovaj bi mu poslao sledeći znak. Štampač će mikroprocesoru dojaviti srećnu činjenicu de je završio ispisivanje korišćenjem takozvanog *interapta* (prekide)

Ako ste nekada razkloptli vaš računer i poglededeli mikroprocesor videli ste de se radi o jednom crnom kućištu se mnogo nožica. Jasno je da svaka od tih nožica ima po neko funkciju — neka omogućavaju adresiranje memorije (adresna bus iti adresna megistrale) neki prenošenje podetaka (dete bus iti megistrela za podetke) neki odrediju de li je u tom trenutku potrebno učitevanje podetka u registar iti njegovo upisivanje u memoriju (R/W) i slično. U ovom trenutku nas posabno interesuju pinovi koji su obeleženi kao NMI *(Non Maskable Interrupt)* i IRQ *(Interrupt Request)*

Pošto je izvršio neku instrukciju mikroprocesor će obavezno testireti logičko stanje nožice obeležena se NMI. Ukoliko je ona ne potencijelu od oko 5 V neme namsakiranog interapta i mikroprocesor mirno prelazi na detje ispitivanje. Ukoliko međutim konatatuje da je nožica NMI ne potencijalu od oko 0 V nastaje prava mele uzbuna trenutni sadržaji registara P i PC (tj. stanje svih flegove i adresa naredbe koje bi se sledeća izvršile da nija bilo interapta) se smeštaju na stek e zatim sa u registar PC upisuju bajtovi iz memorijskih lokacije &FFFA i &FFFB — možemo de zamislimo de je periferijski uređej netareo mikroprocesor da izvrši JSR (&FFFB)! Ovaj program trebe de sačuva sadržaja svih registara koje nemerava de koristi na steku (P registar je već autometski sačuvan) da uradi posao zbog koge je pozvan vrati registra u ranija stanje i izvrši instrukciju RTI (ReTurn from tnterrupt). Posle tope sa nastavlja prekinuti program koji uopšte na zna de je bio prekinut. Nemaskireni interept se obično koristi ze rad sa brzim periferijskim uređejma kao što su hard diskovi i posebno za komunikaciju se mrežeme računera kod nekih jednostavnijih modete računera NMI se koristi isključivo za prekidenje mrtvih petlji koje su nastale usled greške programera. Sve u svemu upotrebe nemaskiranog interapta je uglevnom rezervisana za konstruktore operativnog sisteme vašeg kompjutera.

Maskirene interepte se druge strane koriste kako konstruktori operativnog sisteme tako i korisnici računare. Pošto je mikroprocesor ustanovio da neme namekairanog interapta ispituje stanje nožice označene se IRQ. Ukoliko ustanovi de se one nalazi ne potencijelu od oko 5 V interapta neme pe mikroprocesar izvršava sledeću instrukciju. Ukoltko je na oveoj nožici logička nula (oko 0 volti) detektoven je maskirani interept

Pošto je detaktovao postojenje maskirenog interapte mikroprocesor ispituje stanje i *(Interrupt disable) flege*. Ukoliko je ovaj fleg setoven interapt je maskiren tj. programer je onemogućio njegovo prthvatanja i mikroprocesor će ignorisati zehtev za prekid i preći na izvršavanje sledeće naredbe. Ukoliko sa druge strana ustanovi da je t resetoven mikroprocesor će sačuvati P na steku setovati I zabranjujući dalje interepte i prači na izvršavanja rutine za obradu prekide. Adrese ove rutine se nalazi u memorijskim ćelijama &FFFE i &FFFF — na samom kraju ROM-e

Interapt rutina trebe kao što znemo de sačuva ave registra koja će koristiti na steku de izvrši obradu koje treba de bude što kreće i nejzad de posle vređenja registera u ranije stanje izvrši RTI čime će novi prekidi biti omogućeni. Primenom naredbe CLI možemo da dozvotimo prekide i ranije u toku izvršavanje interapt rutine (pošto su registri sečuvani) eli se to radi samo u slučajevime da postoji herdver kojt regulše prioritete prekida kod osmobitnih kućnih računera prekidi su skoro bez izuzetka onemogućeni dok se izvršave interapt rutine

Procesor 6502 ime i jeden specijelitet kojim se konsnici Z80 ne mogu pohvaliti: softverski interapt. Kada mikroprocesor neide ne instrukciju BRK (BReaK) poneša se skoro keo de je dobio interapt. P i PC se smeštaju na stak i izvršave JMP (&FFFE). Postoje ipak dve razlike između hardvarskog i softverskog tnterapta. Po nailsaku ne BRK instrukciju mikroprocesor pre svega setuje ili fleg čime interept rutini signelizira de uzrok prekide ne treba tražiti ne periferiji. Osim toge RTI na izaziva povratek ne instrukciju iza BRK obevezno se preskače jeden bajt

Softverski interapti se u sistemskom programiranju uglevnem koriste za prijavljivanja grešaka kada operativni sistem detektuje grešku izvršiće BRK pri čemu će sledeći bajt ROM e biti kod (0- 255) greške koji omogućave interapt rutini da laptop odgovarajuću poruku

Neki jednostavniji računari koriste prekide isključivo za osveževanje internog časovnike i njime atika što znači de prekidi nastupaju u pravilnim vremenskim intervalima na primer 50 puta u sekundi. Složenijim (i skupljii) računarima nije potrebna taraglja mikroprocesora pri generisanju

nema smisla vraćati u glavni program jednostavno PLA PLA će skloniti se steka svaki treg da je potprogram uopšte pozivan Novi sadržaj akumulatore u tom slučaju treba ignorisati

Često je potrebno da se potprogramu prenasu naka vrednosti koja će obrađivati I da se u glavni program vrate rezulteti ta obrade Argumente potprograma ja najzgodnije uplasti u neke od registara Ukoliko potprogram ima više od tri argumenata možete ih upisati u naka memorijske ćelije se tiknim adresama odakla će ih potprogram ,pokupiti u tom slučaju potprogram ne sme da poziva samog sebe ali su vam ovakve tzv mikurzije za sada sasvim napotrebna Moguća je najzad smestiti sve argumante na stek a onda izvršiti JSR Potprogram tada najpre mora da sačuva adresu povratka (npr u ćelijama &2000 I &2001) setim da obradi argumente postepeno ih skidajući se steka, da smesti rezultate na stek I da, konačno izvrši JP (&2000) Sva u svemu prilično komplikovano

U ROM-u vašeg računara se nalazi mnogo korisnih potprograma koji vas oslobađaju briga o pisanju po ekranu skaniranju tastatum generisanju zvuka i slično ser nije lakše napisati JSR CH J u akumulatoru dobiti kod pritisnutog tastera nego ispitivati dirka jednu po jednu? Da bista međutim napisali JSR OSRDCH morate de znate gde se tačno u ROM-u nalazi rutina koja skanira tastaturu (ako ja ona smeštena počevši od adrese &FFE0 kao kod BBC-ja i Electrona na početku vašeg programa ćeta napisati SRDCH=&FFE0) kako treba pripremiti njen poziv i gde su smešteni rezulteti Takve podatke možete da nadete u knjigama tipa The Complete ROM Disassembly ,Advanced User Guida ili u našim umacima Nabavite ako je to ikako moguće litemturu u kojoj ja opisan ROM vašeg računara i videćete da će one posteti najkorisnija referenca u praktičnom radu

Periferija i interapti

Sva instrukcije koje smo do sada upoznali bava se prenašenjem podatka po mamorijl I njihovom jednostavnom obradom Pitenjem odakla podaci u msmorijl sa nismo mogli baviji a obzirom da su obezbeđeni potprogrami u ROM u koji će nam ćak i kada se oslobodimo od bajzika omogućiti primanje podatska I izdavanja mzultata postojanja instrukcija za ulaz/izlaz nse na mora interesovati Varujući, međutim..da će deo čitalaca ovog umatka jednoga dans pokušati da segradi naki mali mikroprocesorski sklop koji je nazavistan od računara..posvetili smo ovo poglavlja komunikaciji mikroprocesora sa spoljnim svetom

Procesor 6502 nema posebnih instrukcija za ulaz I izlaz podataka — komunikacija se obavlja prako memorijske mape Pratpostavićemo da je računar povezan se štampačem I da je štampaču dodaljena memorijska lokacija &FE10 Ukoliko želimo da se na papiru pojavi slovo A izvršćemo program LDA #ASC A STA &FE10

Rakio bi se da ovo što smo radili nama veze se štampačem uplasli smo naki broj u ćeliju čija je sdresa &FE10 Ova je ćelija međutim fiktivna tu na postoji nikakav RAM koji bi primio podatak Umesto toga hsrdvar će dakodirad (prepoznati) tmnutak kada se na sdresnoj magistrali mikroprocesora pojavila adrese &FE10 I podatak se magistrala za podatke prosladiti do nakog odportova

Ss periferije ćemo ponesaći i uzimati podatke se šta koristimo obično LDA Nama naravno nikakvog smisla uzimati podatka se izlaznih portova ako bismo u gornjem primaru posla STA &FE10 pokušali da vidimo šta je upisano u ovu ćeliju (se LDA &FE10), teško bismo tsmo pronašli ASCII kod slova A — podetka treba uzimaji samo sa ulaznih portova Da bismo vas sasvim zbunili mći ćemo da je ponakad istoj mamorijskoj lokacji dodaljan ulazni I izlazni port koristimo dakla STA &FE10 I LDA &FE10 prj čamu nama nikakve veze upisujemo i brojeve koje čitamo!

Ako ponoeno pokušate da primanite novostečeno znanja I napimate program koji će na primar slati podatke posmdstvom raznorasnih memorijakih lokacija dsesće se razna čudna atvari) možds će zvučnik zapištati možda će se na akmnu pojsviti besmislice možda ćete uništiti sqdržaj diskats (ako ste toliki navarovatni baksuz ds pogodita kako se traka formstiraju) možda će se računsr izgubiti ili se na vašu veliku žalost neće desiti baš ništa! Da bista smisleno koristli mamorijski mapiranu periferiju morata de znate određana protokola koja je lokacija povezana se kojim umdajem) na koji način uredaji primaju podstka Morsćete dakle da konsultujete literaturu koju ste dobili uz računar domaće I strane kompjutemke časopise III knjige koje sa nude na stranom tržištu Suvišno je reći da je raspored portova ssavim različit od računsrn do računara) da mu u ovom umatku opšte priroda nećamo posvaćivati posebnu pažnju

Najjednostavnij mahsnizam komunikacija mikroprocesora se spoljnim svatom je takozvani hand shaking (u bukvalnom prnvodu — rukovanja) Kada je gospodin Barimer (čitali ste slike onj koneta interapte se mnage druge radnja, ali interapt koji periodična nastupa I dalje

```
        LDX  #0
rekuj   LDA  &FE11      Čekaj dok printer ne bude spremn
        ASL  A
        BCS  cekaj
        LDA  poruka,X   ntedeći znak poruk u akmulator
        BEQ  kraj       tekst se ispisuje dok ne ti potili  nu 0
        STA  &FEA0      kurnktor ne šalje štampaču
        INX             otnet adre p  tedeći kurik ter n X
        JMP  cekaj
poruka  EQUS "Ov      tnapa"
        EQUB 0
```

slika 36

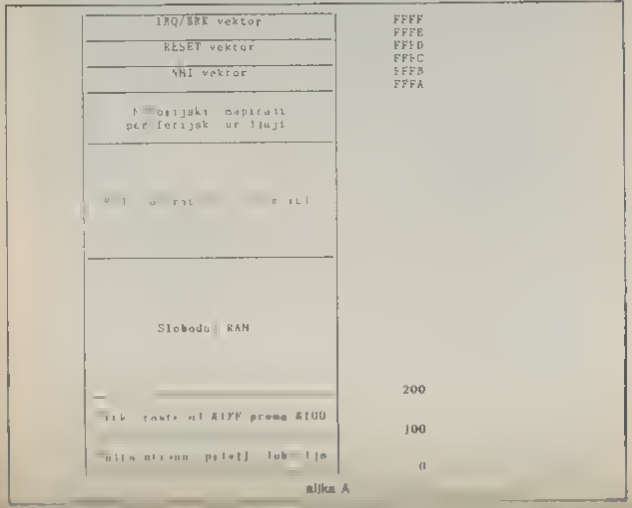
slovima X i Y Svaki od njih može da sačuva po jedan osmobitni broj ali sa avi registri dsleko intanzivnija koriste pri indakalranjima

Registn spacijalna namane se označavalu se PC S I P. PC je šesnaestobitni a S I P su osmobitni registri U setu instrukcija nsčsta naći nijednu jedinu koja operiše se sadržajem registra PC a ipak se taj sadržaj napmkidno menja PC je najma skraćenica od *Program Counter* ili u bukvalnom pravodu programski broječ U ovom sa registru nalazi adrese instrukcija koju mikroprocesor treba da obradi tj adresa bajta koji će pri normalnom izvršavanju programa bln sledeći pročitan Kako sa naka instrukcije sastoja od ivše bajtova registar PC će u toku njihovag izvršavanja nakoliko puta manjati vrednost Pratpostavimo da je ne radu izvršavanja instrukcije LDA &4000 (LoaD Accumulator) koja ja kodirans kao &AD &00 &40 I kao što vidimo zauzima tri bajta memorije &AD ja oznaka instrukcije LDA a &00 I &40 predstavljaju adresu &4000 koja se kao i uvak prikazuje se nižim bajtom napred Pra izvršavanja instrukcije u PC će biti upisane sdress msmorijska ćelije koja ssdrži broj &AD — početak instrukcije koja se obrađuje

Mikroprocesor najpre uzima kod instrukcija (&AD) i smešta ga u registar naradbi koji ja se nas navidljiv Istovmmana se registar PC povećava za 1 tako da pokazuja na bajt &00 Mikroprocesnr pmpoznaja instrukciju LDA i primaćuje da mu je potrabna šesnaestobitna adrese pa sahtava učitavanja njanog nižeg bajta ne koji pokazuje PC U registar adrese (ponovo navidljiv se nas) dolazi broj &00. a PC se povećava se 1 tako da pokazuja na broj &40 Sladi još jedno pristupanje memoriji i prenošenja broja &40 u registar adrese..pri čemu se PC ponovo povećava i pokazuje ne sledeću instrukciju koju mikroprocesor tmbe da izvrši Pra nego što dođe do nje..mikroprocesor će jasno, morati još jednom da pristupa mamoriji da bi pročitao sadržaj ćelije &4000 i smestio ga u akumulator

Iako to u setu instrukcija nije izričito epomanuto možemo lako da upisujemo brojeve u registar PC Instrukcija JMP &3A00 (JUMP) bi slobodno mogla ds se zova I LOAD PC =33A00 Posla izvršavanja ova instrukcije mikroprscasor će nastaviti narmslan rad ali nsćs izvršavati sledeću naredbu već onu čiji je broj ,naslimo upisan u PC — u našem slučaju naredbu čiji je početak upiman u ćelju &3A00 Omogućsne nam je i ds dodajamo konstante ssdržaju mgistra PC — takve dodavanja smo već upoznali pod imanom *relativni skokovi*

Registar S (*Stack pointer* ili pokazivač steka) ima nekih sličnosti se registrom PC njega vrlo retko direktno menjamo (obično se takva operacija obavlje nagda ne početku ROM-a) ali su njegova implicitna promane vrlo česte Kada nam zatrebe ds privmmano sačuvamo sadržaj nakog registra upotrabićemo instrukciju poput PHA (push accumulator) Tada će sadržaj akumulatora biti prekopiran u memorijsku ćeliju na koju pokazuje S a onda će S biti umanjan se 1 kako bi pokazivao na mamorijsku ćeliju u koju će biti upisan sledeći broj koga budemo slali na stek.



Suprotnu proceduru će izvršiti naredbe PLA (puti accumulator) koja će kao krajnji rezultat preneti u akumulator broj se vrha steka i povećati S za jedan. Umesto PHA i PLA sa stekom operišu i naredbe PHP i PLP koja su zadužena za čuvanja statuse svih flagova na steku.

Vidimo da je stek na neki način suprotan programu: register PC se povećava kada obradimo instrukciju što znači da program raste od nižih ka višim adresama. Suprotno tome sadržaj S se smanjuje kada stavimo broj na stek što znači da ovaj raste ka nižim adresama.

Pre nago što nastavimo da opisujemo registra moramo da objasnimo jednu naoko neobičnu stvar vezanu za sjek rekli smo da *osmobitni* register S pokazuje mamorijsku lokaciju u koju će biti smešten podatak iz akumulatora kada primamimo instrukciju PHA. Osam bitova je maduimm sasvim nedovoljno da se ukaže na neku mamorijsku ćeliju: mamorijskih ćelija ima 2 16=65536 dok se sa osam bita može odabrati samo jedna od 2 8=256! Stek se po konvenciji mikroprocesora 6502 smešte u memorijska ćelija čija su adresa 256—$12 (&100—&1FF) — na prvu stranu mamorija.

6502 ja mikroprocesor koji vrlo jasno deli mamorju na stranice veličina 256 bajta. Šesnaestobitnu adresu neke ćelije možemo naima da podatimo na dva osmobitna dela: prvi će specificirati jednu od 256 stranica a drugi jednu od 256 reči na toj stranici. Tako adresa &213C označava šezdesetu ćeliju (63C=60) na trideset trećoj (&21=33) stranici memorije. Strane i (&100—&1FF ili 256—511) je rakli amo rezervirana za stek strana nula (&0000—&00FF) je kao što ćemo videti posebno privilegovana dok se poslednji bajtovi strana &FF koristi za određena vaktore o kojima ćemo još govoriti. Ostala strana mamorija su sasvim ravnopravne i dala se na prostor koji pripada ROM u (obično poslednja strana memorija) i na RAM. Na slici 6 ja prikazana principijalna memorijska mapa računara koji koristi 6502.

Moguće je da ćemo pišući prva programa u nekoj petlji greškom izvršiti mnogo PHA instrukcija koja će biti po bajt prepuniti stek. Kada se register S naprakidnim umanjivanjem spusti do nula njegova će vrednost ponovo postati &FF pa će sa stek kružno popunjavati uništavajući svoj prethodni sadržaj. Ova situacija obično ne dovodi do potpunog kraha sistema (kod računara Z80 će mnogo PUSH-ova izbomberodavati kompletan RAM i uništiti program) ali svakako onemogućava normalan završetak mašinskog programa i rad se potprogramima.

Došli smo najzad do registra P koma ćemo posvetiti dužnu pažnju. Već smo rekli da mikroprocesor poseduje određen broj flegova koji se koriste za kontrolu toka programa i neke aritmatičke i logička operacija. Sva ovi flegovi su ukhvno svrstani u registar P kako bi njihova stanja mogla da se sačuva na steku naredbnom PHP (push P) i docnije restaurira naredbom PLP (puti P). U daljem tekstu ćemo pobrojati sve flagove registra P i opisati njihovu funkciju za sada ćete samo povrhno pročitati ovaj opis ali ćete se na njaga svakako vraćati kada budamo dali primara upotrabe flegova.

Bit nula P registra se zova „Carry" i naravno obeležava slovom C. Ovaj bit se setuja kada sa pri binarnom sabiranju *pojavi* prenos iz najatarijeg razreda (sabiramo na primar $33 i $FA pokušajte) ili kada sa pri oduzimanju na *pojavljuje* posejmice (oduzimamo manji broj od većega). Postojanja ovoga flaga kao što ćemo videti omogućeva jednostavno sabiranje i oduzimanja brojeva koji su smeštani u nakoliko mamorijskih ćelija. C fleg, isto tako predstavlja daveti bit akumulatora pri šiftovanjima pa omogućava pranošenje bitova iz jednog broja u drugi.

Bit jedan regattra P ja bar za početnika najznačajniji i najredje korišćeni flag obeležavamo ga sa Z i zovemo *Zaro flag*. On sa automatski setuja kada je rezultat naka operacije nula ili kada je test dao rezultat True (tačno) i kao tekav ja neobično koristan za realizaciju struktura koje odgovaraju bejzik naredbi IF A=B THEN.

Bit diva se obeležava sa I i zove *Interrupt disable bit*. Programar ga satuja kada želi da njegov program na buda prekiđen zahtevima sa periferije (o interaptima ćamo još govoriti). Savatovali bismo vam da za sada na dirate ovaj bit jer će njegovo trajno setovanja varovatno onemogućiti normalan rad vašeg kompjutera.

Bit tri *Decimal mode flag* ili D se koristi za izbor binarnog ili binarno kodirenog decimalnog moda rada. Kada saviđete osnova programiranja na 6502, pomoći ćemo vam da upoznate vrednost ovoga tlaga za sada će on stalno biti resatovan.

Bit četiri *Break flag* ja za vse još manja značajan on omogućeva razlikovanja apotjnog i unutrašnjeg interapte. Njima ćemo se beviti na samom kraju ovog umatka.

Bit pet je uvek setovan — na koristi se.

Bit šest registra P ja najzad značajan označava se V (od *Overflow*) i lma funkciju koju nije teško razumati setuja se ako je razultat prathodnog sabiranja ili oduzimanja nakoristan uared prekoračenja (pogledajte uvodno poglavlje i sabiranja označenih brojeva) a resatovan ako je rezultat koraktan.

Znajući da bit sedam avakog bejta nesi znak broja, nećete sa iznanaditi kada pročitate da se bit 7 registra P zova *Negative flag* i obeležava se N. On je satovan kaoa je rezultat naka operacije negativan tj. ako mu je najviši bit setovan. Ovaj tleg se koristi za malizaciju mašinskih struktura koja odgovaraju bejzik naredbi IF A<B THEN.

Prenošenje podataka

Prva grupa instrukcija mikroprocesora 6502 kojom ćemo se beviti su naredbe koja su zadužena za pranošenja podataka izmedu mamorija i registara. Neka od tih instrukcija smo već upoznali u desadašnjim primarima amo često koristili nešto poput LOAD A #20 što ja trabelo da znači sećamo se obuhvata sve flegove). Pogledajmo najpra kako radi PHA (PUSH A)

```
STA $100+S
DEC S
```

Ovaj program na bi radio ako bista ga otkucali jer 6502 na poznaje ni jednu od dva korišćene instrukcija. I pored toga on ročito pokazuja sve što se dešava: akumulator se preosuja u ćeliju čija ja adresa **na prvoj strani** data sadržajam registra S a zatim se ovaj registar umanjuja za jedan. Kada nam nakoliko instrukcija docnija bude potrebno da vradimo podatak sa steka u akumulator izvršićemo PLA (PULL A) odnosno

```
INC S
LDA $100+S
```

Šta se dešave ako napišemo PHA PHP? Na stek najpra biva smeštun sadržaj akumulatora a zatim sadržaj P registra. Što znači da za vraćanja podataka u registra trabe koristiti PLP PLA **a ne** PLA PLP. Zbog toga se stek naziva i LIFO strukturom (LIFO=last in first out) poslednji podatak koji stavimo na stek će biti prvi koji ćemo primanom PULL. pročitati.

Jasno ja da izvršavajući PLA mikroprocesor na može da zna da li je bajt koji pronađe na steku nakada bio u akumulatoru on se tamo mogao nači i posle PHP. To nam daje način da dovedamo sadržaj svih flagova u akumulator ili što ja mnogo zgodnija, jednom jedinom PLP instrukcijom postavimo sve flegove u željeno stanja. Preporučujemo vam lpek da se avanturama popuit ova bevitta tek kada seviadate osnova mašinskog programiranja.

Često će vam zatrabeti da na steku sačuvate sadržaje svih registara mikroprocesora. Obzirom da na standardnom 6502 na postoja instrukcija PHX PHY...PLX i PLY (one su obezbedene tek na 65C02) moraćete da se poeluživte malim trikom. PHP PHA TXA PHA TYA PHA. Vidimo da amo na stek najpra posiali sadržaj status registra i sadržaj akumulatora a da amo zatim X i Y registar slali u akumulator da bismo ih onda stali na stek. Vraćanja registara u početno stanja postižemo sakvencom PLA TAY PLA TAX PLA PLP. obratite pažnju na redosled!

Šte se dešava ako se u nakoj petlji nade vide instrukcije PUSH nego PULL? Stek će sa postepeno punjti a registar S će smanjivati sve dok na dođe do nula. Po sledećem amanjrvanju u S će sa naći broj &FF i stek će sa puniti kružno — podaci sa efektivno gube bez mogućnosti povratka! Ovakva će greška izazvati krah programa ali je varovatno da po pritisku na RESET memorija neće biti izbomberdovana — prepunjana steka ja uništilo samo sedržaje ćelija &100—&1FF.

Kada u bejziku napišete GOSUB 1000 računar će početi da izvršava potprogram koji počinja od linije 1000. Kada u deljem redu naide na RETURN vratiće se izvršavanju segmenta programa koji se nalazi iza GOSUB 1000. Da bi to uradio on na neki način mora da „zna" da ja do linije 1000 stigao uz jedan poziv potprograma (inače će RETURN izazvati grešku RETURN without GOSUB.) I da „zapamti" gde se nalazila naredba koja je posvela potprogram. Ovi podaci se zarsta i pamte ali su sa proseičnog programara nabitna njihova lokacija.

6502 ja opremljen instrukcijom JSR *(Jump to Subfloutine)* koja je potpuni akvivalant GSUB razlika ja jedino u tome što se iza GOSUB piše broj programske linija a iza JSR memorijska adresa ćelije u koju je upisana prva instrukcija potprograma (uz korišćenja esamblara će se jasno iza JSR naći ima labela). Za povratak u glavni program koristi se RTS, potpuni akvivalant RETURN.

Da biste znali sva o potprogramima reći ćemo da nama preprieka da u toku izvršavanja nekog potprograma pozovete novi potprogram a iz ovoga sledači i tako do prilična dubina.

Kada biamo ovima zaključili poglavlje o potprogramima znali biste koliko i bejzik programar koji mimo koristi GOSUB ... RETURN na razmišljajući o tome kako ova struktura funkcioniše. Mašinski programer **mora da zna više** pe ćemo pokušati da prikažemo delovanja instrukcija JSR i RTS.

Kada naide na RTS 6502 **mora** da zna gde da se vrati to jest gde sa nalazi instrukcija koju trebe da izvršava. Obzirom da ja svaka instrukcija određena dvobejtnom adrsaom ori nalseku na RTS su potrabna samo dva bajta. Gde bi ta dva bajta mogla da budu smeštena? Pa jedini ozbiljan kandidat može da buda stek.

JSR nn je otprilika akvivalantno sa PUSH PC JP nn a RTS sa PULL PC (jasno je da instrukcije PUSH PC i PULL PC pod tim imantma na postoje). Pri svakom pozivu potprograma na atak dakle bivaju smešjana dva bajta koja posmatrana kao šesnaestovni broj predstavljaju adresu instrukcija na koju će se preći po naliasku na RTS. U principu bismo mogli da napišemo LDA PHA LDA a zatim RTS — mikroprocesor bi se vratio na instrukciju koja ja upisana u memorijaku ćeliju &3001 (zašto na &3000? Na pitajta pravila — mali hir konstruktora 6502) bez obzira na činjanicu da prathodno nije izvršeno nikakvo JSR. Tako amo, premda to nema mnogo smisla, simulirali JP &3001!

Daleko je problematičnija situacije u kojoj smo u potprogramu koji je pozvan sa JSR nehetice upotrebili nako PUSH koje nije imalo svoje PULL. Naišavši na RTS, mikroprocesor slapo uzima dva bejta sa steka i smešte ih u PC na znajući da se na radi ni o kakvoj povratnoj adresi već o podatku koji ima naku drugi smisao. Izvršavanje programa će se u tom slučaju nastaviti od nekog nepredviđenog mesta što može da ima katastrotalna posledice po program u memoriji. Sva u svemu **eke se planira povratak iz potprograma** upotrebe steka mora da bude čista. Koliko podataka stavimo na atek toliko moramo i da uzmamo i tako u svakoj grani potprograma!

Zašto amo rekli: ako se planira povratak ? Kada u bejziku izvršimo GOSUB **moramo** da izvršimo i RETURN. Inače se (naima nevidljiv) atak za pazive potprograma nalako punt pa će jednom biti prijavljena greška tipa Too many GOSUBs. Ne bi imalo nikakvog smisla učiti mašinski jazik ako bi t on imao slične ograničenja ako u nekoj grani potprograma konstatujemo da se

```
                    10111101 / 101 = 00100101
                    101
                    ---
                     011
                      101
                     ----
                     1001
                     -101
                     ----
                      100
```

slika 33

praćeno odgovarajućim šiftovanjem količnika. Ako vam ovaj postupak izgleda prekomplikovano pogledajte sliku 34 — posle proučavanja primera sva izgleda mnogo jednostavnije. Pošto smo razvili metod pisanje programa sa slika 35 na predstavlja naročiti problem — trebe sa samo podsetiti neredbi INC i DEC koje omogućavaju povećavanje i smanjivanje sadržaja neke memorijske ćelije za 1

Korak	Akumulator	Deljenik	Količnik
Start	00000000	00101111	00000000
Shift 1	00000000 <== Ne može se oduzeti 101	01011110	00000000
Shift 2	00000000 <== Ne može se oduzeti 101	10111100	00000000
Shift 3	00000000 <== Ne može se oduzeti 101	01111000	00000000
Shift 4	00000001 <== Ne može se oduzeti 101	11110000	00000000
Shift 5	00000010 <== Ne može se oduzeti 101	11100000	00000000
Shift 6	00000101 <== Može da se odnme 101	11000000	00000000
Oduzimanje	00000000	11000000	00000001 (1 podatak količ)
Shift 7	00000001 <== Ne može se oduzeti 101	10000000	00000010
Shift 8	00000010 <== Ne može se oduzeti 101	00000000	00000100

slika 34

```
101 / 10 = 100 (ostatak 10)
```

slika 35

```
3000          deli
3000 A9 00    LDA #0
3002 A2 08    LDX #8                  ;brojac bitova = 8
3004 38       SEC                     ;Carry se setuje pre prvog LSR odne
3005          sledeci
3005 0E 13 30 ASL deljenik
3008 2A       ROL A                   ; Ratiranje da se "nhvati" prenos
3009 CD 1C 30 CMP delilac             ; Da li je deljenik develjne
300C                                  ; veliki za oduzimanje?
300C 90 06    BCC dalje               ; Nije develjna veliki - dalje
300E E8 1C 30 SBC delilac             ; Oduzmi delilac
3011 EE 14 30 INC deljenik            ;     i dodaj 1 kaliciuku
3014          dalje
3014 CA       DEX                     ; Umanji brojac
3015 D0 EE    BNE sledeci             ;     i ponovi operaciju 8 puta
3017 8D 1D 30 STA ostatak             ; Ostatak u memorija
301A 60       RTS                     ; Natrag u bejzik
301B 00       deljenik FCB$ 0 *       ; Mem- celija koja nave deljenik
301C                                  ; a docnije i rezultat
301C 00       delilac  FCB$ 0         ; Mem  celija koja nave delilac
301D 00       ostatak  EQU$ 0         ; Mem  celija koja cuva ostatak
```

```
10 PRINT
20 INPUT "Prvi broj  ";A
30 INPUT "Drugi broj ";B
40 POKE deljenik A
50 POKE delilac, B
60 CALL deli
70 PRINT "Rezultat  ", PEEK(deljenik)
80 PRINT "Ostatak   ", PEEK(ostatak)
90 GOTO 10
```

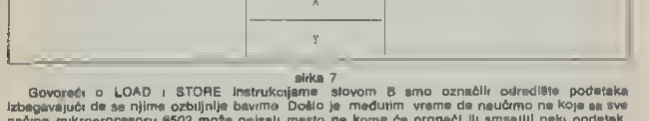
Potprogrami i stek

Stek je rekli smo područje memorije koje se koristi za privremeno odlaganje podataka. Na stek primanom instrukcije PUSH možemo da smestimo akumulator ili P register (P register *dovedi broj &20 u registar A*. Da sada budemo malo egzekinji u mnemonici &502 naredba LOAD se piše kao **LDm** ® i znači *napuni (LOAD=napuni) registar m sa sadržajem B*.

Malim slovom m smo označili bilo koji od registara A X i Y što znači da postoje instrukcije LDA LDX i LDY koje omogućavaju prenos podataka iz memorije u registre

Poiraban nam je jasno i način da prenosimo podatke iz registara u memoriju za jo se koriste naredbe STm B (STore=upiši) gde je sa m označen registar (A X ili Y) a sa B odredište podatka. Tako će STA &2000 smsaiir sadržaj akumulatora u memorijsku ćeliju čije je adresa &2000 dok će STX &613C prepisati sadržaj registare X u ćeliju &613C

Obzirom da posedujemo samo iri registre konstruktorime &502 nije bilo leško da obezbede instrukcije za prenos podataka izmadu njih. Opšti oblik ovih instrukcije je Tmn gde je T skraćenica od Iransfer (=prenesi) dok su m i n oznake registara između kojih se vrši prenos. Podaci puiuju iz **registra m u registar** n što znači da će TAX preneti sadržaj akumulatora u indeks registar X dok će TXS preneti sadržaj registare X u registar S (pokazivač steka). Nije na žalost moguće prenositi podatke izmadu bilo koje dva registre pa će vaš asembler prijaviti grešku eko upotrebite naredbu TXY kojom biste preneli broj iz registra X u registar Y. Kako da zapamtite koje su instrukcije prenosa podatska dopuštena? Najbolje je da pogladate sliku 7 na kojoj je prikazan opšti lok podatake u &502. Moguće je prenositi podatke jedino iz susednih registare S I X X I A A i Y sve su druge kombinacije nadozvoljene!



slika 7

Govoreći o LOAD i STORE instrukcijame slovom B smo označili odredište podataka izbegavajući da se njime ozbiljnije bavimo. Došlo je međutim vreme da nešćmo na koje se sve načine mikroprocesoru &502 može opisati mesto ne kome će pronaći ili smestiti neki podatak upoznaćemo dakle *modove adresiranja*. Pažljivi čitaoci ovog umetka će se dosetiti de smo o modovima adresiranje već govorik u drugom poglavlju eli ponavljanje fundamentalno važnih stvari (na primeru LOAD instrukcije) nikada ne može de naškodi!

Najjednostavniji oblik adresiranja je kao što smo videli implicitno adresa nema tačnije ona se podrazumeva iza naredbe TAX ne nevodimo nikakvu adresu u semom je kodu naredbe određeno da se podatak prenosi iz *registra* A u registar X

Često ćemo koristili i neposradno (*„immediate“*) admeiranje broj se kojim se operiše je smešten iza kode instrukcije. Kod &502 se neposredno adresiranje označava oznakom # LDA #&20 će u akumulator dovesti konstantu &20 (32 dekadno). Semo se po sebi razume da neposredno adresiranje ne može de se upotrebi kod svih naredbi. Še bi značilo STA #&1000 kada se u **konstantu #1000** ne može smestiti nikakav broj (konstanta se tako zove zato što je nepromenjiva)? Konstruktorr &502 su se međutim potrudili de obezbede naposredno adresiranje kade god ono ime smisla. Što znači de ćemo konstante moći da koristimo i kod sabiranje oduzimanje logičkih operacije i ne mnogim drugim mseima.

Apsolutno admeirenje takođe nije teško razumeti — naredba LDA &2000 će u akumulejor dovesti sadržaj ćelije &2000 dok će STA &2100 smestiti ovaj broj u ćeliju &2100. Deiji komentar leško da je potreban.

Obzirom da još nismo upoznali instrukcije skoka, ne možemo da napišemo program koji će preneti sadržaje memorijskih ćelija &2000 &2001 ... &20FF u ćelije &3000 &3001 ... &30FF. Pokušajmo ipak, da razmislimo kako bismo napisali takav program pa ćemo videti da je tako nešto skoro nemoguće — potrebne su nam instrukcije koje će pri svakom prolazu kroz petlju prenositi podatke između **različitih** memorijskih ćelija i akumulatore. Tako u igru ulaze indeks registri X i Y

Pogledajmo instrukciju STA &2000 X primer *indeksiraneg admeiranja*. U početku će na primer X biti jednako 0 pa će mikroprocesor upisati #10 u ćeliju čija je adresa #2000 + 0 = #2000. U sledećem će prolazu X postati 1 pa će #10 biti upisano u ćeliju broj &2001. Posao će se ponavljati sva dok X ne bude imalo vrednost &FF kade će se pristupiti ćeliji &20FF. Še će de se desi kade mikroprocesor sledeći put uveća vrednost X za jedan? Obzirom da je X osmobitni registar &FF je najveći broj koji može da se upiše u njega pa će sledeće INX (*increment X* ili u slobodnom prevodu *povećaj X za jedan*) dati 00 umesto &100 što znači da će se upisivanje kružno ponavljati. To je izbegnuto primerom BNE (*Branch if not equal tj. skoči ako je različito*) instrukcije u programu sa slike P. ciklus će se ponavljati sve dok se sadržaj registare X različit od nule.

Vaš obzira na je što nam program sa slike 8 ne može biti de kraja jasan pre nego što upoznamo aritmetiku skokova i reći se flegovima trebalo bi da primetimo problem koji se ne bi mogao rešiti

```
        LDX #0              ; X pokazuje čeliju strane &20
                            ; u koju treba upisati podatak
        LDA #10             ; Uplauje se broj 10
        uplsivanje
        STA &2000 X
        INX                 ; X pokazuje sledeću ćelliju…
        BNE uplsivanje      ; kada X pređe 0  dođi
                            ; ovo da počeste ćelije
        RTS                 ; Povratak u kejzik
```

slika 6

primenam indeksiranog adresiranje ije da amo želeli da premestimo blok veći od 256 bajta? Obzirom de se u indeks registar može upisati samo osmobitni broj i de je bazna adresa (ono &2000 u STA &2000, X) fiksna. Indeksirano adresiranje nam na može pomoći da realizujemo rad se blokovima u kojima se nalazi više *od* 256 bajta podataka?

Upoznajmo de blamo rešili problem *indirektno postindeksirano adresiranje* koje ćeme prvučiti na primeru instrukcije LDA (&20) Y Pošto se radi o relativno komplikovanom adresiranju najpre ćemo upoznati obično *indirektno adresiranje* koje bi kada bi na 6502 postojalo ilustrovala instrukcije LDA (&20) (ova instrukcija, de to naglasimo još jednom na **postoji** na standardnom &502 ali će nam pomoći da shvatimo **postojeće** LDA (&20) Y)

Kad apsolutneg adresiranje se uz svaku naredbu navodi adresa memorijeka ćelije u kojoj se nalazi podatak koma treba pristupiti LDA &20 bi u akumulator dovalo broj iz memorijaka ćelije čije je adresa &20 LDA (&20) je bitno različita instrukcija u njoj pše de u akumulator treba doveali podatak **čije je adresa** smeštena u ćeliji &20 Mikroprocesor će dakle najpre morati da pristupi ćeliji &20 de pročita iz nje adresu a de zatim pristupi ćeliji čiju je adresu upravo dobio i u akumulator smesti njen sadržaj Memorijska ćelije &20 je jasno nedovoljne da referencira bilo koju od 65536 ćelija pe će LDA (&20) podrazumevati de se adresa sastoji od brojeva u ćelijama &20 i &21 Možde je ave ovo lekše razumeti na primeru naka je u ćeliji &20 bej 0 au &21 broj &41 tada će LDA (&20) u akumulator dovesti sadržaj memorijske ćelije čije je adresa &4100

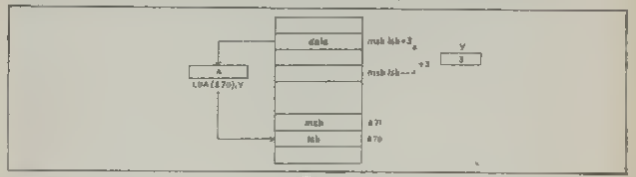
Ako emo razumeli princip delovenje instrukcija LDA (&20) neće vam biti teško de razumete primenu instrukcije LDA (&20) Y umesto da se zadovolji adresom koju je pročitao u ćelijame &20 i &21 mikroprocesor će na nju dodati sadržaj registra Y De se vradimo ranijem primeru sekvence se slika 5

```
        LDA #0
        STA &20         ; dovedjenje broja  00 u ćellju &20
        LDA #&41
        STA &21         ; dovedjenje broja &4i u ćellju &2i
        LDY #&10        ; dovodjeuje broja &10 u registar Y
        LDA (&20) Y
```

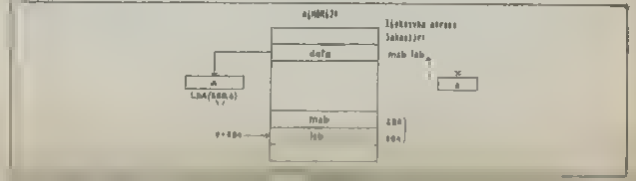
slika 5

će dovesti sadržaj memorijske ćelije &4110 (&4100+&10) u akumulator

Princip delovenje *indirektnog postindeksiranog adresiranja* je dat na slici 10

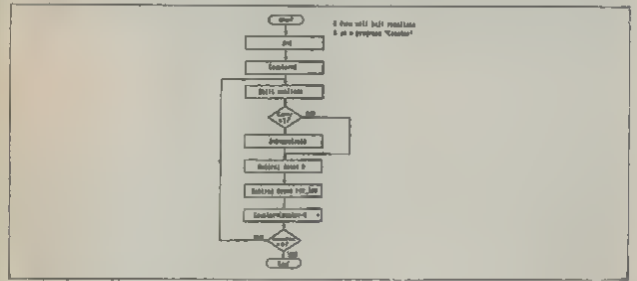


slika 10



Imitiranje manuelnog množenje može da razuituje blok dijagramom se slika 30 — po izvršavanju programe koji je određen ovim dijagramom dabili bismo proizved MPD i MPR u ćeliji PP Primećujemo međutim i određena problemе množenjem dva osmobitna broja možemo da dobijemo broj koji ima 16 bita i koji preme tome, na može de stane u jednu memorijaku ćeliju; algoritam se slika 30 će u tom slučaju dati pogrešan rezultat, a neće signalizirati grešku. Jednostavnim ispitivanjem bismo bez mnogo problema detektovali grešku ali su nam apetiti nešto veći — hoćemo da sastavimo program koji množi osmobitna brojeva i deje šesnaestobitni rezultati

Do rešenja dovodi samo jedna dobra ideje umesto da šiftujemo množenik, šiftevaćemo parcijalni proizvod i te kroz dva memorijska ćelije posredstvom carry flega. Na slici 31 vidimo blok dijagram…e na slici 32 program koji obavlje koraktno množenje osmobitnih brojeva Ako uspete de pročitate ovaj umetak do kraja, upoznaćete metod za množenje šesnaestobitnih binarnih vrednosti



slika 31

slika 32

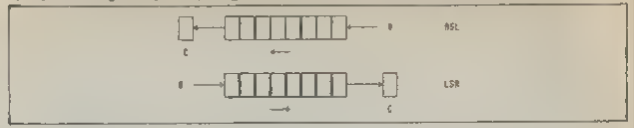
```
3000            mnozi
3000 A9 00      LDA #0                  ; MSB odgovara pr
3002 A2 08      LDX #8                  ; Brojac bitova = 8
3004            sledeci
3004 4E 19 30 LSR mnozilec              ; Sledeci bit masziuc
3007 90 04      BCC ratiraj             ; Bit je jedinica
3009 18         CLC                     ;    mnozenik se d
300A 6D 1B 30 ADC mnozenik              ;
300D            ratiraj
300D 6A         ROR A                   ; Najnizi
300E 6E 1A 30 ROR rezultat_jow          ;
3011 CA         DEX                     ;
3012 D0 F0      BNE sledeci             ; Da l
3014 8D 1B 3D STA rez ltat high         ; Ako
3017 60         RTS                     ;
3018 00         mnozenik        EQUB 0  ; Nem
3019 00         mnozilac        EQUB 0  ; Nem
301A 00         rezultat_jow    EQUB 0  ; Nem
301B 00         rezultat_high   EQUB 0  ;

10 INPUT "Prvi broj ";I
20 INPUT "Drugi broj ";J
30 POKE mnozenik,A
40 POKE mnozilac,B
50 CALL mnozi
60 PRINT 256*PEEK(rezultat_high) + PEEK(rezultat_low)
70 GOTO 10
```

. . . i deljenje

Dok se binarno sabiranje svodi na sukcesivna dodavanje deljenje ekvivalemo višestrukim oduzimenjem delioca od deljenika kao na slici 33 na kojoj emo delili brojeve 109 i 5 (%1011 i 101) i dobili količnik 35 (%100101) i ostatak 4 (%100)

Kao što možete da pretpostavite program koji bismo napravili simulirajući , ručna računanje na bi bio naročito dobar Umesto tega šiftovećemo sukcesivna bitove deljenika u akumulatoru ave dok na budemo mogli de mu oduzmemo delilac Tada ćemo dodati i količniku i nastaviti se radom ave dok ne budu šiftovani svi bitovi deljenika Svaki šiftovanja deljenika mora da bude

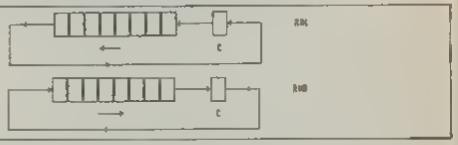
poslednji bit „preliva" u carry indikator. Rotiranje je unekoliko slično tim što se na kraj ćelije na upisuje nula nego ranije stanje flega C. Da vidimo kako zva to izgleda na primerima



slika 27

Logičko šiftovanje nalevo i nadesno je prikazano na slici 27. Vidimo da se ASL i odnosno LSR i pomeramo biteve odredišta i nalevo odnosno nadesno za jedno mesto, umesto nuitog (za ASL) odnosno sedmog bita (kod LSR)...u odredište ulazi nula dok bit koji „lepadne" usled pomeranja odlazi u carry fleg. Reniji sadržaj carry flega se jednostavno gubi.

Vidimo da smo primenom instrukcije ASL odnosno LSR izgubili bivše stanje carry flega - u odredište upisali jednu nulu. Zašto da ne izbegnemo gubitak tog bita i veštačko generisanje nule prepisavši prethodno CY na kraj odnosno početak odredišta? Tako dobijamo operacije ROL i i ROR i rotaciju odredišta i i carry flega za jedno mesto u levo odnosno u desno. Oke ova operacije su prikazane na slici 29

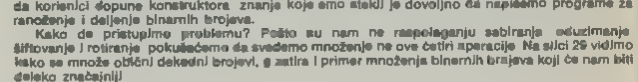


slika 29

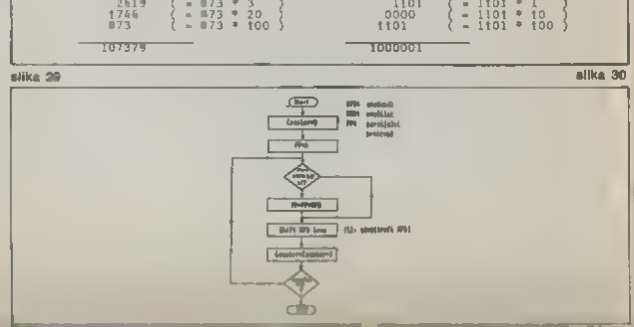
Množenje i . .

Mikroprocesor 6502 poput većine osmobitne sabrata, nije opremljen instrukcijama za ranoženje i daljenje brojeva, leko su ove operacije u praktičnom redu neobično potrebne. Ostalo je da korisnici dopune konstruktora znanja koje emo stekli je dovoljno da napišemo programe za ranoženje i deljenje binarnih brojeva.

Kako da pristupimo problemu? Pošto su nam ne raspolaganju sabiranje oduzimanje šiftovanje i rotiranje pokušećemo da svedemo množenje ne ove četiri operacije. Na slici 29 vidimo leko se množe obični dekadni brojevi, g zatim i primer množenja binarnih brajeva koji će nam biti daleko značajniji



slika 29 slika 30



Patpunosti radi ne slici 11 dajemo princip delovanje *indirektnog preindeksiranog adresiranja* savetujući vam de priču o njemu preskočite i ostavite za boije vremene ovo sa adresiranje retko koristi pa će vas nepotrebno zbunjivati

Indirektno preindeksirano adresiranje ćemo upoznati ne primeru instrukcije LDA (&20 X). Neka sa u X registru nalazio broj &10. Mikroprocesor će najpre ovej broj sabrati sa &20 i tako dobiti &30. Zatim će pristupiti ćelijama &30 i &31 i tamo pronaći šesnaest bita naka adrese: ako sa u ćeliji &30 nalazio broj nulg a u &31 broj &11, pristupiće sa memorijskoj ćeliji čija je adresa &1100 i njen će sadržaj biti prenet u akumulator. Indirektno preindeksirano adresiranje se obično koristi za red sa takvieme podataka koji su rsauti po memoriji, ali au njihove lokacije upisane u sukcesivne memorijske ćelije.

Pošto emo opisali skoro ave moćove adresiranje pristupačne ne 6502 posvetićemo sa njihovim ograničenjima. Sve instrukcije koje srao u ovora poglavlju pominjali su postojeće i mogli biste da ih koristite u svojira programima. Ukoliko biste se mađutim odvažili da napišete LDA (&2000)..Y vaš bi asembler sa zadovoljetvom prijavio grešku. Odakle greška kada ste napisali savršeno korektnu instrukciju koja koristi indirektno pcatindeksirano adresiranje? Jednostavno konstruktorime 6502 je odgovaralo da kod pra i post indeksiranog adresiranja adresu smeštaju isključivo u memorijske ćelije nulte strane!

Nulte strane *(zero page)* obuhvata memorijske ćelije 0-255 (0-$FF) koje su kao što rekosmo, ked &502 veoma privilegovene (svuda ta privilegije). Keko, zaboge, neke memorijske ćelije mogu de budu privilegovene? Asemblerajmo instrukciju LDA &2000 i LDA &20 prva će biti kodirane kao &AD &00 &20 (zauzima tri bajta) a druga kao &A5 &20 (samo dve bajta)! Kako je jedan bajt ušteđen? Konstruktori 6502 su imali dsaie slobodnog mesta u satu instrukcije pa su odlučili da (ato) neradbi (LDA) dodele jeden kod ukoliko sa argument nalazi ne nultoj strani a drugi kod ukoliko je on ne proizvoljnom mestu u memoriji! Ukaliika suter RDM-a nekog računera ne nultu stranu smeti podatke koji će sa najčešće koristiti, uštedeće dosta prostora i dobiti u brzini rade kompjutera!

Deleko je važnija druga privilegija memorijskih ćelija nulte strane koju emo već pomenuli u okviru pre i post indeksiranog adresiranje **memorijske ćelije koje sadrže adresu mogu da se nalaze ne nultoj strani.** To znači de ćete, ćim požeiite da koristite indirektno adresiranje, morati de konsultujete uputstvo za upotrebu vašeg kompjutera i saznete koje su ćelije nulte strane ostavljene korisnicime, ukoliko sa umešate u prostor bajzik interalara moguće su razne neprijatne posledica!

Dalje ograničenje je obaveza korišćenja registra X kod praindeksiranog adresiranje i registra Y kod postindeksiranog. To praktično znači da **ne smete** da napišete LDA (&20 Y) odnosno LDA (&20), X **morate** da koristite LDA (&20 X) i LDA (&20) Y

Dok je za prethodne ograničenje teško naći neko racionalno objašnjenje (što na znači de ono ne postoji!!), prilično je jake razumeti da ne **postoje** instrukcije LDX (&20 X) LDY (&20) Y LDX &2000 X i slične ukoliko je sa adresiranje potreban registar X razuiitat sa njkako ne može istovremeno smeštati u njega (hardveraši će reći da tekozvani master-slave flip-flopovi omogućavaju baš takve trikove ali je to ovde sasvim nebitno)

Na kraju umetka dajemo spisak instrukcija mikroprocesora 6502 i u okviru njega adresne rasdovi koje sveka od nerađbi dozvoljeve nije fola da pogledate tatelu i pokušete da ilustrujete sve ano o čemu smo govorili u ovom poglavlju. Jer ovo je najteže ali i najvažnije poglavlje naše Škole programiranje na 6502 ukoliko ste ga dobro razumeli ostatak ćete pročitati i primeniti sa lekočem! Ukoliko vam nešto nije jasno, ne očajavajte prečitajte još jadnom prathodne stranice a onda jednostavno nastavite sa radom — kade vam god buda potrebno neko egzotičnije adresiranje lako ćete pronaći i u Prakai razumeti njegov opis!

Aritmetičke operacije

U uvodnom poglavlju emo upoznali operacije sa binarnim brojevime koje 6502 može da izvrši sabiranje i oduzimanje. Rekio bi sa de o ovim operacijame ne treba ranogo govoriti: šar ime nečega jednostavnijega od sabiranje? Pa ipak činjenica da 6502 operiše isključivo sa camobitnira podacima nas neiaruje da upoznamo trikove koji će nam omogućiti da operišemo sa brojevime većim od 255 ili +127

Naredba za sabiranje se u mnamonici 6502 označeve sa ADC B gde je sa B označena lokacija broja koji traba dodati sadržaju akumulatora. naredba LDA #&20 ADC #&30 bi, ne primer irabale de dovedu broj &20+&30=&50 u akumulator. Nije mađutim, sve baš tako jednostavnin skraćenica ADC je izvedena od *ADd with Carry* što znači de se pn sabiranju razultatu dodaje i stanje Carry flega. Potrebno je dakle da najpre razuramo keke sa to dodavanje izvodi pa de onde odgovorimo ne pitanje **zašte** je ono potrebne. Pogladajmo dva jednostavna primera sa slika 12

```
CLC             SEC
LDA  #&20       LDA  #&20
ADC  #&30       ADC  #&30
```

slika 12

instrukcije LDA u oke programe iraba de dovedu konstantu &20 u akumulator posle čega sledi dodavanje broja &30 sadržaju ovega iagsiire (razultat sa kae što verovatno pretpostavijete ponovi smešta u akumulaioj). Pitanje je naiavne što će sa pe izvršavanju erogrsme dabiti u

akumulatoru. U akumulator će uvek stići broj &20+&30+C gde smo sa **C** označili stanja carry flega. Kakvo je to stanje? Obratimo pažnju na prvu naredbu drugog programa SEC ova reč je skraćenica od *Set Carry flag* što bi moglo da se prevede i kao *stavi 1 u C (carry)*. C će dakle biti 1 pa će izvršavanjem drugog programa u akumulatoru biti &20+&30+1=&51! U prvom programu smo sa CLC *(Clear Carry* odnosno *obriši carry)* stavili nulu u C indikator što znači da će se po izvršavanju programa u akumulatoru naći broj &20+&30+0=&50!

Šta bi se desilo da smo izostavili naredbe SEC odnosno CLC? Pa kao bi se da je tada stanje carry tiga nedefinisano ali ono jednostavno ne može da bude takvo tiag može da bude ili setovan ili resetovan trećega nema! Zato će izvršavanjem prathodnog dela programa (ili radom bazik interpretera ako ja naš mašinski program uprevo pozvan) biti formirana vrednost C pa će naredba ADC raditi korektno iako možda na onako kako smo želeli. Zapamtiteta da ako planirate korišćenja naradbi ADC i SBC (subtract oduzimanja) **uvek** setujte carry primanom SEC ili ga resetujte naredbom CLC

Videli smo kako se izvršava naredba ADC sli nam je i dalja najsenо kakev je smisao kvarenja tepo dobijanog zbira dodavanjem carry flega. Zamislimo da treba da seberemo dva šesnaestobitna broja od kojih se jedan nalazi u memorijskim ćelijama &2000 i &2001 a drugi u &2002 i &2003 Zbir trebe da smestimo u &200A i &200B. Kako da rešimo problam? Pokušaćemo da najpra seberamo sadržaja mamorijakih ćelija &2000 i &2002 (manja signifikantna bajtove jli kako se to kaže LSB *(low significant byte)* obe broja) i smestimo rezulte) u &200A a da zahm seberemo sadržaja &2001 i &2003 (MSB odnosno most *significant byte)* i rezultat smestimo u &200B. Program koji to radi je dat na slici 13

```
1 LDA  &2000    ; LSB prvog broja u akumulatur
2 CLC
3 ADC  &2002    ; sabiranje sa LSB drugog broja
4 STA  &200A    ; LSB rezultata u ćetiju &200A
5 LDA  &2001    ; MSB prvog braja u akumulatar
6 CLC
7 ADC  &2003      sabiranje sa MSB drugog broja
8 STA  &200B    ; MSB rezultata u cetiju &200B
```

slika 13

Isprobajmo ovaj program stavljajući u &2000 i &2001 broj &2030 (POKE &2000 &30 POKE &2001 &20) a u &2002 i &2003 broj &4050. Pčela startovanja programa dobićemo da se u ćelji &200A nalazi broj &80 (ako iskoristite PRINT PEEK (&200A) dobićete 128 što ne treba da vas iznenadi 128=&80) a u ćelji &200B broj &80 što znači da je zbir &8080 program dakle radi ispravno. Ne šta ako u ćelije &2000 &2001 &2002 i &2003 upišemo respektivno brojeva &20 &3A &F0 i &10? Po startovanju programa će se dobrti zbir &4A10 dok je &3A20+&10F0=&4B10. Program je dakle pogrešan!

Kako je nastupila greška? &502 je najpre sebrao brojeva &20 i &F0 i dobio ili bar trebao da dobije &110. Broj &110 je, međutim veći od &FF pa se pojavio bit koji nije mogao da stane u akumulajor. Onako kako smo ga napisali program je zanamario taj bit pa je sebrao brojeva &3A i &10 i dobio &4A da je dodao pranos dobio bi &4B što bi bio tačen rezultat.

Da je &502 jednostavno zanemario bit prekoračenja problam sabiranja šesnaestobitnih brojeva bi bio veoma teško rešiv. Na svu sreću konstruktori mikroprocesore su se pobrinuli da se prekoračenja prenese u Carry indikator koji na neki način predstavlja deveti bJi akumulatore. Da vidimo kako ava to izgleda na primeru se slike 14

```
                C
&20 =       00100000
&F0 =       11110000
ADC =   1   00010000
```

slika 14

Shvatanje činjenica da se deveti bi rezultata sell u Carry indikator će nam pomoći da brzo i lako rešimo problem sebiranja višebajtnih brojeva. Dovoljno je da izostavimo instrukciju broj 6 iz programa koji emo sastavili pa će prenos jednog sabiranja biti pridodat rezultatu drugoga. Kako se prenos na engleskom zove carry jasno vam je po čemu je C tiag dobio ime — posredstvom njega se prekoračenje u jednom sabiranju dodaje na rezultat drugog sabiranja kake bi se formirao korektan višebajtni rezultat.

Naredbe ADC čeim na C utiče na Z, V i N flegove ukoliko je na primer rezultaj sabiranja 0 biće setovan Z(ero) fleg. Kako može da se dobije nula sabiranjem dva pozitivna broja? Pokušajte na primer da seberete &F0 i &10 i dobićete &100 kako u akumulajor može da stane svega osam bita rezultata u njega će biti upisano 00 uz setovanja zero (i carry) flega. V fleg signalizira da je rezultat u akumulatoru nakorakjan ako sa argumanti poemajraju kao označeni brojevi. Naprevimo dakle mali rezima pri sabiranju naoznačenih brojeva (0-255) setovan carry fleg označava prekoračenja i nakorektan razultat. dok se V fleg obično zanamaruje. Kod sabiranja označenih brojeva (-128 do +127) prakoračenja se signalizira sejovanim oVerflow tiagom pri čemu se stanja Carry tiega obično zanamaruje.

Pretpostavimo da se u ćelji &2000 nalazio broj &9A odnosno %10011010. On je najpre dovedan u akumulator a zatim je na njega i broj %01111111 primanjena operacija „I". Na slici 23 vidimo šta se dobija

```
                10011010
AND             01111111
                --------
                00011010
```

slika 23

Vidimo da se rezultat razlikuja od početnog podatka &9A samo u poslednjem bitu koji je posle primana AND operacija postao nula. Ovakav rezultat se mogao i očekivati broj %01111111 se sastoji od samih jedinica od kojih svaka kada je logički pomnožimo (operacija i se zova i logičko množenja) sa bilo čim drugim daja kao rezultat to nešto drugo. Jedino vodeće nula kada ja logički pomnožimo sa bilo čim daje kao rezultat opet logičku nulu!

Šja bismo učinili da ja trebalo da **setujeme** sedmi bit ćelije &2000? Upotrebili bismo naredbu ORA kao na slici 24

```
LDA  &2000    Davodjenje sadržaja ćelije &2000 u akumulajor
ORA  #&80     Setovanje sedmog bita
STA  &2000    Saeštanje rezultata u memoriju
```

slika 24

Uzevši da se pre izvršavanja programa u ćelji &2000 nalazio broj &73 odnosno %01110011 novi sadržaj je prema slici 25 jednak &F3 — sedmi je bit očito setovan

```
                01110011
OR              10000000
                --------
                11110011
```

slika 25

Da li emo mogli da testiremo nakj bit mamorije a da ga na promanimo? Naravno da emo mogli — trebalo je samo primaniti program sa slika 26. Vidimo da smo najpre prepisali broj iz ćelija &2000 u akumulator a zatim ga logički pomnožili se %00000010. Ukoliko je prvi bit (sećate se da se broji od nultog?) akumulatora jedinica rezultat će biti %00000010 ako je taj bit bio nula u akumulatoru će se naći nula. Znajući da naredba AND deluje na Z(ero) tiag koji će biti setovan ako se po izvršanju operacije u akumulatoru pojavi nula naredbe *BEQ nije* će nas odvesti do segmanta programa koji je označen se nije samo ako prvi bit memorijska ćelije &2000 nija setovan

```
LDA &2000       Ćelija &2000 u akumulator
AND #%00000010  Test b   je prvoj bita
BEQ resetovan   Ni li je 0
setovan

                Nar ibi koj   e i          j

<R>
res t var

                Narcibe koje se i vršavaju ako   e

RTS
```

slika 26

Konatruktori 6502 su se pobrinuli za jednu specijalnu instrukciju koja nam pomaža da latovramano testiramo šesti i sedmi bit neke mamorijska ćelije to je naredbe BIT. Koja se uglavnom, koristi za testirenja zauzetosti neke perifarijska jedinica. Ukoliko na primer, izvršimo BIT &2000 akumulator će biti uporeden se ćelijom &2000 (kao da smo iskoristili AND) ali se sadržaj akumulatore neće promanitl! Da bi stvar bila još lepša sedmi bit memorijska ćelije &2000 će biti prenesen u fleg N a šesti bit u fleg V.

Ekskluzivnom se disjunkcijom u ovom momantu nećemo previše baviti. Njena zanimljiva osobina je da je sama sebi inverzna: ako izračunate *(X EOR Y) EOR Y* gde su X i Y bilo koji brojevi dobićete opet X. Ukoliko dakle sve memorijske ćelije u nekem segmentu ekskluzivne pomnožimo se nekim brojem dobićemo nazke tesmisleh sadržaj koji ponavljajući ovu operaciju vrećamo na normalu. Mašineka naredba EOR je dakle šesj gesi razniih zašićenih programa.

Šiftovanje i rotiranje

U uvodnom smo poglaviju ukratko objasnili pojam šiftovanje sadržaj akumulajara ili neka ćelije se pomera za jedan bit ulava ili udesno pri čemu na neki od krajeva ćelija dobija 0 a se

Instrukcije grananja za razliku od JMP instrukcija zauzimaju po dva bajta memorije) Mikroprocesor ja najpre iz memorije uzeo kod instrukcije BEQ ($F0) i prepoznao ga a zatim pročitao adresu $20 U tom trenutku registar PC pokazuje na memorijsku ćeliju $1002 u koju ja upisana naredba koja će biti izvršena ako Z fleg nija setovan Ako je flag setovan mikroprocesor će sabrati sadržaj registre PC ($1002) i broj $20 pe ča tako dobijanu vrednost stavlti u PC Izvršavanja programa će tako biti nastavljano od Instrukcija čija je adresa $1020

Šta ja dobijeno a šta izgubljano ovakvim adresiranjem? Pre svega program u koma koristimo iskljućivo relativno adresiranje će koraktno radili ma u koji ga segmani mamorija učitali BEO PC+$20 izaziva preskakanja sledeći $20 bajiova. baz obzira gda ja ova instrukcija smeštenal Trebe međutim_da kažemo da je na 6502 skoro namoguće a i sasvim napotrebno pisati programa koji koraktno rade bez obzira na avoju lokaciju u memoriji — dalako ja lakše izmanjti ORG i asemblirati program na drugo mseio Relativni skokovi se zato kod većina mikroprocesore konsta iskljućivo radi ušteda memorijskog prostora kod kratkih akokove koji au inače dalako češći od dugih JMP $2020 zauzima iri a BEO PC+$20 semo dva bejta Za šta je onda nakoma poirabno apsolutno adresiranje? Korišćenjem jednog jedinog bajte se pamćenja rastojanja skokovi su ograničeni na 127 bajta (broj iza _branch instrukcija se posmatra kao označen što znači da au mogući i skokovi unazad) dok iza JMP možemo da navedamo bilo koju adresu u čitavoj mamorijskoj mapi

Konstruktori 6502 su na žalost učinili da zamana Instrukcija JMP sa _branch buda problamatična nije obazbeđen bezuslovni relativni skok — naredbe BRA *(BRanch Ahvays* pomalo smešno zvuči ali — to ja mnamonika) sa pojavila tek kod docnijih verzija mikroprocesora 65C02 U većini se slučajeva na sreću može računati se stanjam nakog od flegova pa čemo tako „JMP kraj zamanti se BNE kraj ako smo **sigurni** za Z fleg na može da bude setovan Ovakva ušteda međutim na preporučujemo početnicima i iskusan će programar ponakad pogrešiti Ako se već odvažite na slične avanture **obavezne** propratite lažna BRANCH Instrukcije nakim komantarom — kada docnije budete gledali program svekako ćeta se pitati _kakvo je ovo iestrenja ?

Većina asamblara tretire relativna akokove kao I apsolutne pisaćete BEO DALJE baš kao što biste pisali i JMP DALJE Sam asembler će pronaći labalu OALJE I izrečunati njano rastojanja od lokacija Instrukcija BEQ formirejući iako korakatan adrsen dao ove naredbe

6502 omogućava i indirektna apsolutna akokova KMP ($2000) će na primar Izazvati akok na nairukciju **čija je adresa** upisana u ćelija $2000 i $2001 Ukoliko su na primar u ćelija $2000 I $2001 upisani brojevi $1A I $F0 JMP ($2000) će izazvati skok na adresu $F01A

logičke operacije

Došlo je vrema da se pozabavimo *logičkim* operecijama AND ORA I EOR To su binarna operecija koja imaju po dva argumenta i koja se koriste neobično često dovoljno da im posvatimo posebnu pažnju

Jedan od argumanate bilo koja od ovih operecija se nalazi u akumulatoru u kojl se smešta i rezultat Drugi može da se nalazi u okviru sama instrukcije (AND #'n neposredno adresiranja npr AND #$F0) Ili u mamoriji (npr AND $2000 Ili AND ($30) Y)

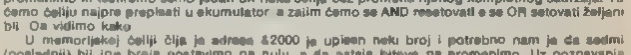
Nije leško pogoditi šta reda naredbe AND a i ORA s sadržaj akumulatore se bit po bit poredi sa a, pe se formire razultat prema tabelama se slika 5 EOR s radi nešto sasvim slično pri čemu se nad bitovima izvodi akaluzivna disjunkcija koja ja definisena tabelom sa slika 21 Ekaklizivna disjunkcija kao što joj i ima kaže liči na disjunkciju (ORA) ali se od nja malo i razlikuja 1 ORA 1=1 ali ja EOR 1=0!

X	Y	X eor Y
0	0	0
0	1	1
1	0	1
1	1	0

slika 21

Za šta može da se iskoristi koja od ovih operacija? Ponekad će nam bili potrebno da promanimo ill testiremo samo jedan bit neka ćelija bez promena njenog kompletnog sadržaja. Tu ćemo ćeliju najpre prepisati u akumulator a zatim ćemo se AND resetovati e se OR setovati željeni bit Oa vidimo kako

U mamorijskej ćeliji čija je adresa $2000 je upisen neki broj i potrebno nam ja da sedmi (poslednji) bil tog broja postavimo na nulu a da ostala biteve na promenimo Uz poznavanja funkcija i najmačemo jednostavan program sa slika 22



slika 22

Pošto smo, na jedvite jade ahvatili kako se koristi naredbe ADC trebe da kažemo šta mogu da budu njani argumanti Jedan od sebiraka je uvek u akumulatoru u koji se smešta i razultat Iza ADC navodimo adrseu drugog sabirka koji može da buda konstanta (ADC #n npr ADC #$20) ili adresa naka mamorijska ćelija Adresa kao i do sada može da bude apsolutna Indeksirana indirektna preindeksirana i indirektna postindeksirana. pri čemu Instrukcija ADC ima posebna kodova za označavanjs skraćenih adresa na nuitoj sireni Sumarni prikaz ove instrukcija je narevne dat u tabeli instrukcija

Oduzimanja ja relativno slično sabrenju SBC a će od akumulatore oduzati sadržaj a pri čemu e može da buda broj (SBC #n) Ili adresa naka mamorijska ćelija (SBC $2000 SBC ($20) Y SBC $3000 X ill nešto slično) U akumulator se primanom ova Instrukcija smešta broj (**A-a-(1-C)**) što izgleda prilično čudno dok za sabiranja dva _normaina broja (de bismo dobili da je 2 + 2 jednako 4) iraba **resetovati** carry flag (CLC) pre oduzimanja dva normaina broja (da bi 4-2 bilo 2) ga traba **setovati** (SEC) U ovom slučaju carry fleg označava pozajmicu pa se zato u literaturi često označava kao *carry/borrow flag* tj indikator koji, zavisno od operecija označava pranos ili pozajmicu Da bismo oduzeli šesnaestobitni broj u ćelijama $2000 i $2001 od broja u ćelijama $2002 i $2003 upotrabićemo dakla program se slika 15

```
LDA   $2002
SEC
SBC   $2000     A=A ($2000) (1 C)  C=1 ako je manji broj oduzet od većeg
STA   $2004
LDA   $2003
SBC   $2001
STA   $2005
```

slika 15

Kao i kod sebirenja Z fleg biva setovan ako je rezultat u akumulatoru nula dok se V setuja ako je nastupilo prakoračenja pri sebiranju označenih brojeva N(egative) fleg označava znak rezultate koji jesno ima smisla samo ako operišemo sa označenim osmobitnim brojavima

Specijalan slučaj sabirenja i oduzimanja su instrukcija INm odnosno DEm gda m može da bude bilo koji od indaka registara X odnosno Y Instrukcija INm (=increment povaćaj) će povećati sadržaj m za jedan tok će ga OEm za isto toliko smanjiti Slično toma Instrukcija INC i DEC omogućavaju povećanje ili smanjanja sadržaja neke mamorijske ćelija se jedan tako da će INC $2000 povećati osmobitni broj u ćeliji $2000 za jedan Videćemo da au ova naredbe vrlo značajna u petljama gde je potrebna često promana sadržaja raznih brojača. INC, INX INY DEX I DEY na deluju na carry fleg što znači da na ovaj način nije pogodno povćavati brojače koji zauzimaju više bajta memorije Ili da je pri tom povećanju potrebno koristiti Z fleg koji se jasno setuja ako je rezultat operacija jednak nuli Zanimljivo je takođe, da Instrukcija koja bi automatski inkrementirala akumulator na **pastoji**, što znači da ćete morati da izvršite CLC ADC #1 da biste povećali akumulator za jedan.

Bezuslovni i uslovni skokovi

Kada bejzik interpretar vašeg računara izvrši naku instrukciju on automatski prelazi na izvršavanja sledeće i tako dalje — sve dok ne naide na neko STOP III ENO Korišćenjem GOTC naredba se narevno može zahtevati da se ovaj tok izvršavanja izmani dok se IF THEN ELSE konstrukcijom postiže izvršavanja sledeće naredba samo ako su neki ulovi ispunjeni Bez ovakvih naredbi na bismo mogli da sastavimo čak i najjednostavnija programa što znači da nešto odgovarejuće mora da buda raspoloživo i na mašinskom jeziku

Najjednostavnije naredbe bezuslovnog skoka ja JMP — potpuni akvivalani GOTO Iza JMP se nalazi adresa mamorijske ćelija u koju je upisana naredba koju sledeću ireba izvršavati što zna či da JMP $3000 zauzima tri bajta memorije Naredba JMP $3000 jsano podrazumeva da se Instrukcija na $3000 zaista može izvršiti kako mnogs instrukcija zauzimaju dva tri ili čak četiri bajta može da se dogodi da se u memorijskoj ćelji $3000 na nalazi početak naka instrukcije već njan drugi ili treći bajt Dok bejzik interpreter u kontlikinim situacijama prijavljuja grešku mikroprocesor na poseduja apsolutno nikakav način da prosudi da li se na $3000 nalazi instrukcija koju sla vi planirali za izvršavanja on će pokušati da izvršava ono što tamo nađe, što će dovesti do nepredvidljivog toka događaja I moguće potpunog blokiranja računara

Iako 6502 razuma jedlno insirukcije tipe JMP $3000 za nas je ovakvo korišćenje apsolutnog adresiranja sasvim nepogodno Pre svega ratko su nam potrebni skokovi na ovako okrugle lokacije — obično programirmo prelazak na neku instrukciju u programu Pri pisanju tog programa nam je jseno poznata njegova početna adresa u memoriji (pišemo ja iza ORG) ali nam nije poznata početna adresa svaka njegove instrukcija da bismo je našli moramo da izračunamo koliko bajta memarija zauzima svaka njoj prethodna Instrukcija pa da tako dobijena brojeva saberemo i dodamo na vrednasi ORG Takvo računanje i sabirania ja naprijatan posao podložan grešakama koji da stvar buda još gora moramo da ponavljamo kada god u programu napravimo bilu kakvu izmenu koja će produžiti ill skratiti neki njegov segmeni Zbog toga svi iola dobri asemblari omogućavaju rad sa *labelama*

Reč labela (label) može da se prevede kao oznaka radi se o nekom simbolu kajim na vam je ni segmeni programa A obleži ali omogućavaju da ima labela buda bilo

koja reč od 5—6 slova sa izuzetkom mnemoničkih skraćenica naredbi ako iskoristimo labelu JMP računar je neće razlikovati od instrukcija JuMP! Jasno je da ćemo ima labela biće ji tako da asocira na svrhu segmenta programa koji označavamo često imena labela su RADNJ RACUN DALJE GRESKA KRAJ i slično

```
3000 A9 00     LDA #0
3002 85 70     STA &70
3004 A9 00     LDA #0
3006 85  1     STA &71          Celije &70 i &71 predstavljaju broj 0000
3008 A0 03     LDY #0
300A              ciklus
300A B1  0     LDA (&70) Y      Celija u akumulator
300C 91  0     STA (&70) Y         a onda  nazad
300E C8        INY              pristupa se sledecoj celiji iste strane
300F D0 F9     BNE ciklus       da li je dostignut kraj strane?
3011 E6 71     INC &71             ako jeste  predji na sledecu stranu
3013 4C CA 30  JMP ciklus       ponavljaj operaciju u beskonacnost
```

slika 16

Na slici 16 je prikazan program koji koristi naredbu JMP da bi realizovao beskonačnu petlju ovaj program upisuje u svaku memorijsku ćeliju njen prethodni sadržaj i tako u beskonačnost! Labela CIKLUS označava početak petlje pa je JMP CIKLUS akvivalentno sa JMP $3000 a tim što je JMP CIKLUS dalako univerzalniji jednostavnom promenom direktive ORG ovaj program možemo da asembliramo u bilo koji segmant memorije. Ukoliko se odlučite da otkucate i isprobate ovaj program uzmite u obzir da će on, evenjualno izazvati krah sistema ukoliko vaš računar (što je sasvim uobičajeno) koristi memorijski mapirena periferijska uređaja (više o njima docnijaj i da ćete moći da ga prekinete jedino pritiskom na RESET taster

Labela osim sa skokove možete da konatite i kao zamenu za brojeve Na početku programa možete na primer da napišete RAMTOP=&5CB2 pa će docnija LDA RAMTOP biti akvivalentno se LDA &5CB2 U ovom slučeju korišćenja labela pomaže jedino boljoj čitljivosti programa jer četa kada ge docnije budete gledali zaboraviti da je &5CB2 adrese sistemska promanljiva RAMTOP pa ćete moreti da konsultujete razna tabela da biste razumeli segmant programa koji ste sami napisali

Uslovni skokovi

Instrukcije JuMP izaziva tzv bezuslovni skok — naredbe čija je adresa navedana isa JMP se izvršava u svakom slučaju U preksi ćemo dalako češće koristiti takozvana uslovne skokova koji su kao što smo već rekli omogućeni postojanjem flegova Naredbe BEQ JEDNAK *(Branch if EQual)* će na primar izazvati skok na segmant programa označan labelom JEDNAK seme ake je Z(aro) tlag setovan Ukoliko želimo da se skok dogodi ako je tleg resejoven koristićemo BNE NEJEDNAK *(Branch if Not Equal)* Slično tome BCS predstavlja situaciju u kojoj je Carry flag seljovan a BCC situaciju u kojoj je resejovan BPL označava pozitivan a BMJ negativan znak broja (N flag resetovan odnosno setovan) dok BVS i BVC označavaju setovan odnosno resetoven overflow tleg Za početak će nam kao što vidimo na slici 17 biti dozvoljeni Z i N tlagovi program koji dajemo nalma u memorijsku lokaciju REZ upisuje jedinicu ako je broj u ćeliji POD pozitivan nulu ako je jednak nuli a 255 ukoliko se radi o negativnom broju (mikroprocesor kada se radi o jednobajtnim rečima smatra brojeve 0—127 pozijivnim a 128—255 negativnim uz korišćenja potpunog komplamanta?)

Uslov		Dejstvo na flegove Z	C	N
Registar	vrednost	0	0	1*
Registar	vrednost	1	1	1
Registar	vrednost	0	1	0*

* — samo ako se radi sa označenim brojevima

slika 17

Preućimo malo program se slike 17 i upoznajmo načina na koji se flegovi setuju Naredbe prenosa podataka u registre pre svega postavljaju flegove u zavisnosti od prenesenog podatka LDA &4000 će na primar setovati Z tleg ako se u ćeliji &4000 nalazila nula a resetovati ga ako je tamo pronađeno nešjo drugo Poređenje sa nulom nam je međutim rejko dovaljno želimo da poredimo dve broja šjo rešimo uz pomoć naredbe CMP (CoMPare) Naredbe CMP ima adresni sleo koji odraduja jedan od brojava koja poredimo drugi se obavezno nalazi u akumulajoru

```
3000 A9 00 31 LDA #0d
3002 F0 0E    BEQ nula
3004 10 0E    BPL pozitivan
3006 A9 FF    LDA #255
3008 8D 01 34 STA rez          broj je negativan
300B 60       RTS
300C           pozitivan
300C A9 01    LDA #1           broj je pozitivan
300E 8D 01 34 STA rez
3011 60       RTS
3012           nula
3012 A9 01 31 LDA rez          : broj je nula
3015 60       RTS
```

slika 18

Tabela se slika 18 će nam pomoći da razumamo stanja flegova posla poređenja Ukojiku su brojevi jednaki setuju se Z i C flegovi Ako su brojevi različiti treba da odredimo koji je veći ali to odredivanja nije baš jednoznačno Posmatrajmo na primar sljuaciju kaoa broj &20 iz akumulajore poredimo sa &F0 (CMP #F0) reklo bi se da je &F0 veći Ukoliko međutim ova podatke posmatremo kao označene brojeve &20 će biti veći jar &F0 predstavlja negativan broj — 16 Carry flag je nadležan za prosto logičko poređenja — u datom će slučaju on biti resetovan jar je sadržaj akumulatora manji od sadržaja mamorija N fleg je se druge strana zadužen za arijmatičko poređenje on će biti resetoven jer je akumulator veći od mamorija

Uslov		Neoznačeni	Označeni
1	A < M	BCC manje	BMI manje
2	A <= M	BCC manje_ili_jednako	BMI manje_ili_jednako
		BEQ manje_ili_jednako	BEQ manje_ili_jednako
3	A = M	BEQ i jo	BEQ jednako
		BCS vece	BPL vece
	dalje	dalje	dalje
4	A >= M	BCS vece_ili_jednako	BPL vece_ili_jednako
5	A = M	BEQ jednako	BEQ jednako
6	A <> M	BNE razlicito	BNE razlicito

slika 19

Bajzik poznaje šest jestova (manja_manja ili jednako veće veće ili jednako jednako i različito) koja trabe bez razmišljanja programirati i u asemblaru u toma pomaža slika 19 koju ćete ubrzo naučiti napamat. Zanimljivo ja da se naka ispitivanja na mogu obaviti bez primana dve BRANCH instrukcija ali su se konstrukjori 6502 pobrinuli da najpolrabniji testovi budu jednostavni za programiranje

Osim instrukcije CMP, opremijani smo i naredbama CPX i CPY koje omogućavaju poređenja broeva upisenih u registre X i Y se memorijom Dajstvo ovih naredbi na flegove ja identično se naredbom CMP

Da čitava ova diskusije na bi ostala mrtvo slovo na papiru, pripremili smo mali primer program se slika 20 pretražuja mamorijska ćelija $3000 — $30FF i prebrojeve pozitivna negativna i nula bejtove a zajim ispisuja izveštaj o razultatima ispitivanja Ispisivanja rezultata smo doduše povarili bejziku jer iz mašinca još na umamo da se obračamo tastatuni i ekranu

```
2500 A2 00     LDX #0
2502 8E 02 40  STX pozit        Brojači se postavljaju na nulu
2505 8E 01 40  STX nul
2508 8E 00 40  STX negat
250B              ciklus
250B BD 00 30  LDA $3000,X      Uzmi sledeći [r i
250E F0 08     BEQ jednako
2510 10 0C     BPL plus
2512 EE 00 40  INC negat        [r                  ]vre -
2515 4C 0B 25  JMP ciklus       [                   ]
2518              plus          [                   ]
2518 4C 1 25   JMP ciklus       [                   ]
251B              jednako       [                   ]
251B EE FF 02 40 INC nul        [                   ]
251E EE 02 40  INC pozit        [                   ]
2521              sledi
2521 E8        INX              [                   ]
2522 D0 E7     BNE ciklus       Ako je [            ]
2524 60        RTS              . Povratak [        ]
```

slika 20

Relativno adresiranje

Sva naredbe grananja (skok ako je neki uslov ispunjen) kod mikroprecesora 6502 predstavljaju primera jakozvanog *relativnog adresiranja* Pogledajmo na primar naredbu BEQ PC + &20 i praipeajavimo da se ova instrukcija nalazila u memorijskim ćelijama &1000 i &1001 (videćemo da

Mali oglasi

Ako ne možete da podnesete da drugi nemaju ono što vi imate objavite svoj mali oglas u „Računarima"

Ako ne možete da podnesete da drugi imaju ono što vi nemate, jevite se na neki od malih oglasa u „Računarima"

Prva stvar koju traba da uradite ja da se odlučite da li želite običan ili uokviren mali oglas

CENA OBIČNOG MALOG OGLASA do dvadeset reči je 900 dinara Svaka naredna reč košte još 60 dinara, s tim što oglas ne sme da ima više od 50 reči. Adresa oglašivača se ne račune u cenu

CENA UOKVIRENOG MALOG OGLASA je 900 dinara po visinskom centimetru, s tim što se mogu zakupiti najmanje 32 slovna znaka. Ako se ne iskoristi čitev prostor u jednom redu, računa se broj redova a ne broj znakova Za uokvirane oglase preko 5 cm cena je 1400 dinara po centimetru

Poželjno je da vaš mali oglas počinje sa Prodajem, Kupujem, Držim časove Menjam ... ili nečim sličnim što ukratko ukazuje na sadržaj oglasa

Da ne bi bilo zabune, obavezno naznačite da li želite običan ili uokviren mali oglas, i zajedno sa tekstom vašeg malog oglasa pošaljite i priznanicu o uplati na adresu redakcija GALAKSIJA, BULEVAR VOJVODE MIŠIĆA 17, BEOGRAD, sa naznakom „za male oglase u RAČUNARIMA"

SPEKTRUM

mamo najnovije programe Tražita besplatan katalog Cena kompleta je 600 — din Zvati oko 20 sati
KALIPSO SOFT Crvenog Križa 11 Zagreb, 041/416-183

SCARE CROW PIRATE CO
Nudimo dobre jeftine igre brzu uslugu uz besplatan katalog Pokon SCARE CROW BEOŽ a po narudžbi usluzne programe Komplet 600.— din KNIGHT RIDER DAN DARACH, ACTION REFLEX, MAFIA CONTRACT 2 MIND STONE NINJA MASTER HUNCH BACK 3, KIONAPP STAINLESS STEEL LABYRINT NON ATLANTIC CHALLENGER
Kenan I Aida Selmenović Bratstva Jedinstva 31/2 71000 Sarajevo tel 642-696 (071) Mešković Dejan, Bratstva Jedinstva 41 71000 Sarajevo, 071/612 138

FUTURE SOFT CLUB vam nudi najatraktivnije programe za ZX Spectrum na YU tržištu pojedinačno i u kompletima CAULDRON 2 SPORT OF KINGS NINJA MASTER ROBIN OF SHERLOCK, HUNCH BACK 3 brza isporuka, BONUS programi povoljne cene Zovite nas na telefone 011/401-995 Vladan I 011/402-218 Saša

SPEKTRUMOVCI!!! Jedini koji uz ni sku cijenu programa (80 I 80 din) snima direktno iz Spectruma Garantirano snimak ispravan I nakon nekoliko godina. Uvjerite se! Uz to stari novi I najnoviji programi popusti I besplatan katalog potražite na adresi
MIHAJLOVIĆ BRANIMIR Kalelarska 43 54000 Osijek

ZX Spektrum — izaberite programe iz drugih oglasa i javite cenu programa 80 din tel 030/81-992

ZX Spectrum Fred Abu Simbel Pinball 2 Rambo Gunfright, Popeye Chimera Art Studio Leonarde The Artist, Master File Mega Basic 3 0, Beta Basic 3 0 Tasword II Collectros Pack, Cijena 800 — din Lelekić Z. K. Temišvarska 12 54400 Đakovo

VIDEO FANTASY vam donosi novitete RUPERT (Quicksilva), BEREBOY (Elite) Universal Hero (Xcel 2), Olettre (Dynamic Soft), Figure Chess, Montronix, Force Fighter Rebel Star BICP Superman Road Runner Mer Maid Madness Disc of Death
Sve ove sa kasetom i PTT troškovima samo 1200 din.
VIDEO FANTASY Fruškogorska 31 21000 Novi Sad tel 021/611 706

Oldtimer SOFT — najveći izbor programa za Spektrum — kvalitetni snimci — umerene cene — besplatan katalog sa novembar tražite na 011/438-137 do 15 časova ili pismom na adresu
Miroslav Radosavljević Braće Nedića 2 11000 Beograd

NOVO U KATALOGU

LASER GENIUS
baterija programa za mašinske programera
— asemblari sa ekranskim editorom
monitori sa dekaderom
analizatori mašinskog koda

THE WRITER
profesionalni program za obradu teksta

GARANCIJA ZA SVE VRSTE MEMA
MILOVANOVIĆ LJUBIŠA
Petra Lekovića 57 11030 Beograd
tel 011/556097 posle 15 h

Spektrumovci — najpovoljnije usluge Bilo koji program 75 dinara. 16 programa po vašoj želji + kaseta C90 + PTT — 1966 din Preporučeno obavezno MULTICOPY-jem! Garancija 6 meseci Najnoviji programi DYNAMITE DAN 2 SPINDIZZY 2 ACE NINJA MASTER KUNG FU MASTER DAN DARACH STAINLESS STEEL BIG BEN NIFTY LIFTY PANAMA YOE SACMAN ALIEN HIGHWAY, KIREL GHOST AND GOBLINS BOULDER OASH WORLD CUP CARNIVAL WHO DARES WINS 2 BOUNCER
Milunović Veselin 7 Vojvođanske brigade 52 21208 Sremska Kamenica tel 021/619-991

VRHUNSKI GRAĐEVINSKI PROGNAMI za Spectrum OKVIRI ROŠTILJI, REŠETKE DIMENZIONINANJE TEMELJI ISKAZ ARMATURE i drugi Za radna organizacija i pojedince Besplatan katalog Glno Gracin Kezele 17 51000 Rijeka tel 051/617-281
PINAT SOFTWARE LTD vam nudi najnovije programe za ZX Spectrum Za specijalan jesenji popust besplatan katalog 011/466-995 Nikola 462-844 Gvozden

Spektrumovci CANNIBALS SOFT vam nudi najnovije programe po niskim cenama — komplet 250 programa 60 din. Veliki popusti Pišite za besplatan katalog Ronald Vlatko M Tita 52 54512 Feričanci Stanić Dalibor 17 travnja 2. 54500 Našice tel 054/723-084 ili 711-883

ROM DISASSEMBLY 1700 din (engleski) Mašinac za apsolutna početnika 1000 din (prevod) Napredni mašinac 1000 din. (prevod) Odlične knjige a pored toga i veliki broj programa sa opširnim uputstvima Rakita Dragan Put 8 P Odreda 25/131 21000 Novi Sad tel 021/399-839

QUICKBREAK SOFT-programi za spektrum i C-64 60 d/kom Od najstarijih do supernovih — snimak direktno iz kompjutera garancija 30 dana Katalog besplatan OBS TA kovska 46 32300 Gmilanovac. Tel 032/714-220

VMS PIRAT CO Njegoševa 15/III 34220 Aranđelovac tel 034/851 334 prodaje preko 400 uslužnih programa za spektrum Veliki broj uputstava Više od 1600 programa u kompletima i pojedinačno Svaka sedmica novi kompleti igara Mogućnost pratpiate Snimili smo cene Apsolutna garancija za vrhunski kvalitet snimka Spisak najnovijih igara besplatan Za veliki katalog sa svim programima i uputstvima pošaljite u plamu 200 din

ZX SPECTRUM !!!!
50 različitih kompleta svaki komplet do 20 programa Kasete SONY (TDK od 90 minuta programi poštarina i troškovi pakiranja — samo 2000 din)!!
Malo razmislite i uporedite i sa drugim ponudama Ekspres dostava. besplatan katalog Iztok Stražar Kajuhova 44 61110 Ljubljana. tel 061/453-907 P S Na sveka 4 kompleta (odjednom) peti besplatan!

MACHINE LIGHTNING — 4 X 48 K Paket programa za dizajniranje igara i rad na mašinci (odlican asemblier 90 strana uputstva) LASER GENIUS 4 programa za rad u mašinci (EKRANSKI EDITOR, ASEMBLER MONITOR DIRAGER-ANALIZER 150 STRANA UPUTSTVA) Tu su još BLAST WRITE MASTER OF FICE BETA BASIC 3 0 C jezik Sve sa opširnim uputstvima.
Rakita Dragan put Bosanskih Partizanskih odreda 25/131
21000 Novi Sad 021/399-839

Super jeftino! Spektrumovci programi 60 dinara a 12 programa 600 din. (Ass. Ghost s) telefon 071/524-089 Vujić Predrag Braće Vujičića 5 71000 Sarajevo

Spektrumovci! Nove igre u kompletima. Cena jednog kompleta (12 igara) bez kasete 600 — din Oba kompleta 1000 din.
KOMPLET A CLIFF HANGER, MOLECULE MAN, SEX CRIME GRAND PRIX HOCUS FOCUS ROBIN OF CHERLOCK 1 2 3
KOMPLET B GHOST'a AND GOBLINS BIGGLES 1 I 2 YU BACK TO SKOOL ARENA Adnan Rustanović Trg Solidarnosti Buha 13 71000 Sarajevo tel 071/544-598 pre podne

Nova pravila

Kada smo pre nekoliko meseci uveli mogućnost da se mali oglasi izdiktira telefonom Imali smo pre svega, u vidu hitne slučajeve Čiteocima se, međutim, ovaj način toliko dopao da druge više praktično I ne koriste čak ni kada se radi o veoma dugačkim i teškim ogiasima prepunih imana najnovijih igara. To je previše skupo i za njih (plaćaju visoka telefonska računa) i za redakciju (veliki angažman oko prijema I veliko administriranje — slanje uplatnica I naknadna provera uplata) I veome nepouzdano — u oglasima ima više grešaka Osim toga, primetili smo da pojedini oglašavači nakon izlaska Računara iz štampe za borevljaju na svoja materijalne obaveze prema listu Stoga smo odlučili da pooštrimo kriterijume za prijem malih oglasa telefonom Na ovaj način sa ubuduće mogu dostavljati mali oglasi samo u dve slučaja

- **ako se naručuje oglas da 20 reči, i**
- **ako se naručuje ponavljanje oglasa, bez obzira na njegovu dužinu, iz prethodnog broja**

Svi ostali mali oglasi mogu se dostaviti na jedan jedini način **poštom se priloženom uplatnicom**

Spektrumovci ! ! ! Pažnja ! ! !
Super-najnoviji komplet od 13 igra za samo 900 dinara (pojedinačno 150 din program)
Komplet 34 Kung Fu Master W Cup Carnival The Planets 1 1 2 Comet Game Big Ben, Pyracurse A C E
Komplet 35 Hunek Back 3 Bebby bearing Cauldron 2 Black Arrows, Knight Rider Bernie Ninja Master
Predrag Đenadić O Karakljićeva 33 14220 Lazarevac tel 011/811 206

SILICON DREAMS neprevo delazi sa velikim novostima u poslovima i obrazovnom softveru
SPEED OFFICE (obrada svih mogućih podataka)
LASER GENIUS (pisanje i učenja mašinca postaće igra)
PROJECTOR (obrada podataka uz visoku grafiku)
THE WRITER (fantastičan tekst-procesor)
BLAST 4 0 (eva nema nike)
MASTER OFFICE (8 fantastičnih uslužnjaka)
MEGA BASIC 4 0 (najnoviji)
Vi samo razmislite i nazovite nas Dobićete uslužne programa iz svih oblasti Imamo sve što se može naći u Jugoslaviji Opširno uputstvo imamo za svaki program Živković Velibor tel 021/870-69 Strihar Milan Bul AVNOJ-a 53 21000 Novi Sad

Spektrumovci ! ! ! Pažnja ! ! !
95 najboljih odabranih uslužnih programa na tri kasete-komplete svaki po 1000 din Komplet uslužni 1—37 programa
PASCAL HP45 ASSEMBLER BETA BASIC 1 8 IS COMPILER ME LOOURNE DRAW
Komplet uslužni 2—28 programa
DEVPAC 3 ART STUDIO ILLUSTRATOR MASTER CORY BETA BASIC 3 0
Komplet uslužni 3—22 programa
TURBO LOAD DEVPAC 7.8 MEGA BASIC LEONARDO LOTO
Predrag Đenadić, O Karakljićeva 33 14220 Lazarevac, tel 011/811-206

Ukoliko želite da ovladate svetom računara naučite neke programske jezike PASCAL FORTH Lisp C Mikro Prolog i sve verzije BASIC-a možete naći kod nas Pobednik računara i naučite ga da radi ono što poželite SILICON DREAMS Živković Velibor tel 021/87-069 Strihar Milan Bul AVNOJ a 53 21000 Novi Sad

KOMODOR

Najnovije i najbolje igre 30—130 dinara (Green Beret, Commando Kane Uridium Spindizzy) Pojusti nagrade iznenađenje besplatan katalog Zvonimir Ožbolt Vrt Jagode Truhelka 2 54000 Osijek, telefon 054/31-950

Izuzetne povoljno svakog meseca kompleti najnovijih programa i igara Komplet sadrži 25 programa Kompilat + kaseta + PTT — 1800 din Mihajlović Dejan Mike Alasa 64 Beograd tel 011/180-834

C-64 Komplet 1 International karate 1 1 2 War play Match Day Green Beret Superbowl Macadam Bumper Saboteur (sa spektrumaj Ghosts n Goblins, Golf Biggles 1 1 2
Komplet 2 Way of diger 1 2 1 3 Gladiator Ping Pong 1 1 2 Spellbound Spindizzy, Trust, Cauldron 2 Formula 1
Komplet kaseta 1200 din Aleksandar Obradović Braće Grbića 26 65340 Heroeg-Novi tel 082/43-374

GEOS 2500 din sa disketom, THE NEWSROOM — Kućna nevina 3500 din sa disketom GIGA-CAD ne 3 dvostrane diskete 5500 din PLATINE 64 2900 din sa disketom MEGA DISK program sauzima celu disketu služi za kopiranje originalnih programa sa trake na disk 2000 din. sa disketom DDJIĆ NENAD Pere Kosorića 18 Beograd 11199

Komodor 64 — komplet 15 najnovijih programa + kaseta + 5 poklona — 1900 din Cyroscipe 3 Art studio Rolelsball Knight Games Cyborg Equinox, Mission AD Sex Games 2 Trirning Piray Titanic, Kriket Eduscm RIDI the jet Štefić Ronald Maksima Gorkog 8 42000 Varaždin 042/46-095

COMMODORE 128 sss PROGRAMI
POJEDINAČNO ILI U PAKETU !
CP/M & ost 128
PREKO 50 NASLOVA
NARUDŽBA ZA 24 ČASA
ss (011) 484-329 ss

COMMODORE 64 — 2 kompleta, svaki 15 programa.
Komplet + kaseta — 1200 din oba 2200 din
Sve pojedinačno (Starquake, Uridium Bombjack)
Spisak besplatan Biljan Zvonimir jr A Mihanovića 6, 58000 Vinkovci

LSH — LABORATORIJA ŠOFTWARE/HARDWARE
Nudimo vam veliki broj najnovijih igara i poslovnih programa za commodora 64 ta izradu poslovnih programa po vašem sahtjevu Vrlo po voljni uvjeti kupnje i mogućnost prepiste sa najnovije programa. Najjednostavniji način naručivanja programa Besplatan najopširniji katalog u našoj zemlji
Reset testeri 1300 — sa poštarinom
Naša adresa LSH F Staraja 10 42000 Varaždin

Širok izbor najnovijih programa za komodor 64 WAR PLAY INT KARATE 1 1 2 ONE MAN AND HIS DROIT URIDIUM 1 2 1 3 RAMBO 2 TIME CRYSTAL MERMAID ? COMBAT ZONE GNOST S and GOBLINS SAMANTA FOX SP ROAD TO NOWHERE BIGLES 1 1 2, RASPUTIN WNG DARES WINS 3 SUPER BOWL, SPELLBOUND LEADER BOARD GOLF Na svakih 10 programa 3 poklanjam Tražite besplatan ktalog. Sa sa 013/812-405 i Bojan 013/813-373

Comodore 64 — Superbowl WHAMI, Kane Last V8 RATS Enigma Force 2 Robin of Sherlock Tau Ceti Nexus Summer Games 3 + kaseta — 1700 din. SIMON ŠERG Vilharjeva 27, 65270 Ajdovščina

Commodora 64 supernitovi Miami Vice Space Talaman Beer Burt Boulder Dash 5, Lords of Midnight 2 Cly Morgos Duest for Tyres 3 Ping-Pong 3 Titanic Piracy War Play Pepsye 2

STUDEN-SOFT-SERVICE CBM-64
Evropske verzije najnovijih američkih Top-hitova. Svi najnoviji potpuno razbijeni programi najjednostavnija preslimavanja Top-VERIFY garancija kvaliteta. Adr Studen Ilija III Saša LAMELA 4/8 77000 Bihać tel 077/223 162

ZA COMMODORE 64 komplet. Beach-head II Uridium II Ping-Pong II Bomb Jack, Kawasaki R R Wham Green Beret, Fairlight Leaderboard Superbowl Rugby + kaseta — 100 din. Željko Filipović tel 067/31-210

MARX ANO ENGELS posiduje YIE ER KUNG FU DESERT FOX, MERCANARY URIDIUM TOUR DE FRANCE PING PONG YABA ĐOO, ELECTRA GLIDE V8 Mc GUIGAN BOXING i još bolja. Vaćina cena 200 din Najveće 400 din Popusti do 50%! Super paketi Besplatne igrai Jcalpović S Petrovića 38 21208 S Kamenica

Komodor 128 — najveći izbor CP/M programa (Wordstar Dbase, C Cobol) disketnih (Yerne Superbase Laser Music Maker) kasetnih (Abacus turbo, Einstein) Besplatan katalog
Štefić Ronald Ul M Gorkog 9 42000 Varaždin 042/46-095

SUNNSOFTWARE CLUB C-64 PC-125 CP/M vam predstavlja u svem besplatnom katalogu izbor samo najkvalitetnijih kasetnih i diskctnih programa Više od 2000 naslova među kojima su i oni koje će drugi oglašavati tek kludeg meseca Tel 021/20-179

Noviteti 86 Besplatan kataleg sa opisima. Pojedinačno ili u kompletima od 25—70 programa. Sarno 2000 din MOVIE MAGNETS BABSIT PIE CASTLE KINGDOM VALDCIPEDE? HACKER TIME TRAX RIPPING JARNS BYNTH SAMPLE 3, BASE DEUTSCH GSF STUDIO SPORT CRAZY COMET EQUINGX, STAR INTRDTUNE DEMO 1 LANDING ON TD 18 JS 90 SPLITTING PERSIRALITIES CONCERT WEMBLEY FLOOPY-BONG PIRACY BEER BELLY BURS LAS VEGAS 2 HUMAN RACEG ADV CREATOR RANSOM THE KING CV DAVID TRAP TTI TALISMAN ELEVATOR INC WHEN ODVES CRY COUNTTOWN MIAMI VICE KNIGHT RIDER RMC TITANIC FUCK DEMOK KNIGHT GAMES 1 2 3 4 i 5 Komplet 13 + kaseta — 2500 din Mujalić Asim. Nurija Pozderca 7a 72000 Zenica tel 072/22-566

C-64 najnoviji programi MIKIE PARREBDY NOMAD STREETHAWK, TURBO ESPRIT EXPLODING FIST 2 MISSION A D KNIGHT GAMES PARALAX. Aleksander 011/489 0982

NEW BEST SOFT nudi najnovija programa najjeftinija u Jugoslaviji Cena jednog programa 20—40 din Besplatan katalog Pišite nami Adresa Franić Danijel I L. Ribara 29 55250 Drljevac

C-64 KORONIS RIFT TITANIC NOMAD TURBO ESPRIT MIKIE, ELEVATOR, PS15 MARBLE MADNESS, GREEN BERETS II KNIGHT GAMES, RAID OVER MOSCOW II INFILTRATOR, TAU CETI, BATTLE FOR BRITAIN, ZOIDS TIME TRAX.
Kamplet, kaseta, poštarina i pekten program 1900 din.
Petrović Branko Senjačka 44 11000 Beograd tel 011/650-508

Samanta Fox, Spellbound Gringo Visitora 2 Beret, Macadam bumper Death Wake Mission Elevator Izaberite 10 programa sa 1000 din
Branislav Čobanov P Drapšina 55/1 21480 Srbobran, tel 021/730-364 Janković Gabor Dr Đorđa Šantića 25 21480 Srbobran

C-64/C-128 OLIMPIC SOFT
Najveći izbor hitova po 40 dinara. Besplatan katalog Moguća razmjena. Empire War Play Saboteur Uridium 2 Biggles, Zorro Time Crystals Phantoma Pejović Jovan Olimpijska 9 71000 Sarajevo. tel 071/461-967

Za C-64 program MEGATAPE probija sve zaštite i kaseta samo 1300 din Besplatan katalog najnovijih programa Kolčrun Marijan trg I Kukuljevića 14 41000 Zagreb

C-64 Veliki izbor programa snimanih direktno iz računara svaki program verifikvan (garantovano bez LOAD ERROR) Komplet sa 30—60 programa 2 000 — din Katalog besplatan. Saša Stoletović Đura Đakovića 1 58000 Vinkovci

DISK programi za C-64 Geos Multiplan Disk Wizzard Platine 64 Tasword Chessmaster 2000 Wordstar Fortran Profi Painter Jet, Megadisk, Newsroom Gamemaker
Zoran Milosavljević, ul Miroslava Krleža 26 Stenovo 34000 Kragujevac

GRIN SOFT Sve što imaju drugi imamo i mi ali još bolja Tu su V-visitora Green Berat Uridium 1 i 2 Death Wake Saboteur Spellbound Leaderboard Way of the tiger 1 i 2 i mnogo drugih super hitova. GRIN LEON Đidikovac 8 71000 Sarajevo tel 071/386-62

ALAN SOFT C64 odabrani programi u paketima iz 7 različitih područja
30 pomoćnih programa
30 akcianih programa
30 sportskih
30 arkadnih
30 muzičkih
30 društvenih
20 radio amateri
1 paket + kaseta — 2000 — din
komplet 7 paketa 10.000 — din
Ako ste početnik najlesne Pesekna besplatna pomoć ALAN SOFT 7 travnja 30 56311 Stubnić

Komodorovci! Superkomplet A Tema hawk, Submarine simulation Babeus Hackman Karnage Visitors Boulder Dash 6 NOMAD Saboteur 2 Reset to nowhere froger parker black hole +5 hitova. Kompiat + kaseta + PTT — 900 din
Gera And Teta Software Mihaila đića 35 34000 Kragujevac tel 034/33-814

SNOOPY UNIVERSAL STUDIOS
C-64 PC-128 (CP/M)
NAJNOVIJI I NAJBOLJI
KASETNI DISKETNI PROGRAMI
U KATALOGU SA PREKO 3000 PROGRAMA
MI NUDIMO MNOGO VIŠE!
075/235-666
TINJIĆ ADI
ŠENJAK E/4
75000 TUZLA

Veliki izbor programa po malim cenama sa vaš C64 Tražite besplatan katalog. Hanšek Ivica Marina Divalte 171 54000 Osijek, tel 064/52-378

POLU PIRAT za komodor 64 poseduje sva tita i ostali all prodaje upola cene izaberite iz drugih oglasa svoj komplet. Agbaba Saša, Marijana Gregorana 44 V Beograd. tel 011/417-371

C64 komplet 2 Cyberg Knight Games (5 programa) NOMAD Dragona Lair Ninja Riddlers Dan Streetfawk, Ninja Master za 1500 din sa kasetom tel 011/481-085

KRIMINAL SOFT ima iz Engleske sve što zamislite Svi programi su u turbu i razbijeni Cijena SVAKOG PROGRAMA je 100 — s 30 izabranih 2300 dinara. Milić Ditto i krajiške brigade 10 tel 077/222-955 i Srdić Dejan Martela The 15, tel 077/227-170, 77000 Bihać

C64 u kompletu SHOGUN MER MAID MADNESS GUIDE KORONIS RIFT THE JET CAUSSES DF CHAOS ROAD TO NOWHERE KNIGHT DF DESERT, BLACK HOLE PING PONO 3 BOULDER DASH 5 WINO RIDE THE FALL GUY Komplet za kasetom 1500. — Siniša Nevaković 11143 Beograd Braće Jerkovića 91 tel 011/481-085

YU C S ja najveći i jedini pravi izvor svih najnovijih programa za C-64 PC-128 s od skore i za IBM PC Ako želite svaki novi program literaturu hardware (spec speedos — modemi), servis besplatno korišćenje našeg YU-mailboxa (011/213-036) obratite za nama
— YU C S — Na produ 38 62391 Prevalje tel 062/851-336 (od 19 h)
— YU C S — Cvijićeva 125/20, 11000 Beograd, tel 011/767 269

Komodore Soft nudi za samo 3000 din 120 najnovijih programa. Cena kasete je uračunata u cenu kompleta. Isporuka odmah Ghost s and Goblins Monty on the run BBC emulator The way of tiger 1, 2 i 3 Star Ping-pong Int. Karate 1 i 2 Spellbound. Green Beret. Speed King Beach Head 3 Spindizzy itd tel 011/532-442 Tomišinović Igor Kumodraška 29 11090 Beograd

Za COMMODORE 64 nudimo vam sve kasetne hitove. Da izlaska ovog broja, naši programi, kao što su Jack The Nipper Sweden Erotic World Games (5 Games III), Druid, Druids Green Beret II Ace of Aces Super Cycle Knight Rally 3D Pikado Asterisk III Beatles War Hawk, Airwolf II Fundus Ide biće već zastareli! Ivan Tošković, Cvijićeva 125/20 11000 Beograd tel 767 269/011

NEWSROOMprogram za C64/C128 na 5 strana diskete plus moje 3 diskete plus program za preanimavanje plus kratko uputstvo plus poštarina — sve samo 4000 dinara. Isporuka odmah. Radanović Milorad Ilija Grbića s56 74400 Derventa

COMMODORE 64 — komplet 20 najboljih novih programa+ kasetu i uputstvo za početnika možete dobiti za samo 2200 din Pišite na adresu Šantić Mihevil Slobodana Principa 14 72250 Vitez, ili telefonom 072/71 705

DTP SOFT vam nudi septembar/oktobar 88 u kompletima Komplet A NOMAD Icupe, Cyborg, Knight Rider Spindizzy, Silent Service Split, Personalities Johnny Reb 2 Ping Pong 3 Titanic. Komplet B W A R Paralax Lap Miami Vice, Time Trax, Flight Deck, Superbowl Hacker 2 Leaderboard Ghosts n Goblins.
Komplet +kaseta+PTT — 1600 din.
Sve ovo i još malo više možete dobiti na adresu Vujić Zoran Bul revolucije 87 Beograd 011/426-801 424-533 Dragan i 456-448 Dule

KOMODOR 64 20 igara BEATLES, CYBERUN MDVIE SEX GAMES 2 RAMBO 3 SUMMER GAMES 3 ASTERIX, MIAMI VICE PUSSI SMDUGHII! MDTO GUZZI TEDDY, KIREL MAFIA 2, GARBAGE FLIPER DRUIDS. BATMAN ICARUS POD, CYROSCOPE 3 KNIGHT RIDER KOMPLET+KASETA+PTT — 1299 din. ISPORUKA ODMAH!!! GOJIĆ NENAD PERE KOSORIĆA 16 BEOGRAD 11196

Komodor 64 — najnoviji programi, pojedinačne ili u kompletu. Besplatan katalog, pošton programi Novo Mega-komplet — 1024 programa s kasetama i uputstvima — 40 000 din
Stefić Ronald 42000 Varaždin Ul M Gorkog 9 tel 042/48-085

SHIFT-SOFT — KOMPLET 4 DRUIDS (uđite u magični svet druida) WARHAWK, INDOR SPORT GHOST HUNTER II TRISTAN ANO ISOLDA (iz vremena kralja Artura) SCARAB DNE BITE TWO DEEP HOLLYWOOD ON BUST DRACULA (šmrt) W A R (reklame iz Video Computer Games-a „To be a hero is easu Staying alive is mere difficult). GYROSCOPE 3 PARALLAX BOULDER DASH 5 7 5 (sva borija od bolješ) JOHNNY RES II SERENADA, ART STUDIG (Suveni crnački hit sa ATARI-a i Mc INTDCHsl) 1 C U P S CYBORG, ARCANA (Pravi nastavak STAFF DF KARNATHI) STREET HAWK, ICARUS PEPSI-COLA (specijalna izmenaenjet), GALAXYBIRDS. INVALIDSKI SOCCER(igrači u kolicima golmani za šlekama publika robijaši — oni humor!) Sve ovo +kaseta+PTT=2500 dinara Mogućnost TURBO TAP-a za Vašim imenom za 1000 dinara!!!
SHIFT — SOFT
Vasović Nenad. 11000 Zemun, Dubrevačka 19 011/210-984
Kramenović Goran 11070 Novi Beograd Dušana Vukasovića 74 011/172-234

COMMODORE 64 vam predstavlja GHOST AND GOBLINS GREEN BERET WAR PLAY BEACH HEAD IV INTERNATIONAL KARATE II LEADER BOARD i drugi najnoviji programi — komplet 15 igara + kaseta — 1600 din. Miljković Slobodan, Jurija Gagarina 198/5 11070 Novi Beograd tel 011/168-196

ALAN SOFT C64 PC128 korisnički programi, aplikacije programske uputstva (SH—HS) Sudite uspešnji na ostu i u kući Naša je mamalma.
„PROGRAMI BEZ UPUTSTVA (SH—HS)=!???=D
PROGRAMI SA UPUTSTVOM (SH—HS)=USPEH
Besplatan katalog sa širokim opisom oprograma.
ALAN SOFT 7 travnja 30, 58311 Slobreg

Komodorovci! Samo ovog meseca za 64sace „Rašunara odobravamo super popust se igre. Imamo sve najnovije Dešaberte 50 programa+kasete+poštarina+najknjiga za programiranje. Referonce Guide =5000 dinara.
Pojedinačna cena je najjeftinija u katalogu koji tražite još danas Rek isporuke je 24h Prijatnu igru želi vam
M and S soft
III Sulevar 130/193
11070 Novi Beograd
Tel 011/146-744

Komodor 16, 116, +4 veliki izbor programa animanih direktno iz računara. Kvalitet super cena povoljna Dragan Ljubisavljević 19210 Bor 3 Oktobar 302/9 tel 030/33-941

Komodorovci COnan, ALICE AZTEC NINJA HACKER 2 CAR DESTRUCTION SET, SHOUGUN CORDNIS RIFT, NOMAD, GIOJE (za 4), TITANIC MIAMI VICE + kaseta + PTT — 1550 din. Sve programe imamo! Pojedinačno DRAGON 3 LAIR + uputstva Dutuković Goran Dalmatinska 2 12000 Požarevac. tel 012/21-961 Zoran Stevanović Verdarska 26/17 Požarevac, tel 012/23-540

C-64
Komplet 1
Karmit s Storymaker Art Studio Sex Games II Boulder Dash 7,8 Ronald Rudback. 4th Dimension Asterix, Caesars Travels Illusion Thrust, Astro Pilot, Pro Boxing DK, Iridis Alpha, Fuck Porno W A R
Komplet 2
Mordon s Quest, Icarus, Space Frontier, Edusom Street Hawk, Triming 130, Bigmine Teddy Rollestolebal! Pricks Perne Game. Dr No Legal Sex Arcana Super Boulder Dash cCheesmaster Rambo Trainer Johnny Reb 2
Cena kompleta 2000 din. + kaseta. Amir Tinjić Senjek E/4 75000 Tuzla. tel 075/235-066

Commodore 16 +4 izaberite 10 programa animljenih turbem Kaseta — 3000 din Winter Games (5 delova) Sea Strike Jetset Willy Mr Pyniverse Rock man W S Baseeball Strip Poker Bongo Xargon wars Commando II Fire Ant, Baby Berka Invasion XL 3D Grand prix Čobanov Nestor Nikole Tesla 16 21480 Srbobran Od 8—13h 021/730-161

Commodore 16,+4 20 programi besplatan katalog Commodore 64 128 — nova ponuda od 4500 programa. Katalog je 300 — din a!i poruđbinom vraćam novac. German Šandor Rade Končara 23 23000 Zrenjanin

NEW NOW SOFT podaje veliki broj uslužnih programa i igara za Komodor 64 Cena niske katalog besplatan, usluge kvalitetna. Za narudžbe preko 3500 — din. besplatite kasete.
Srzak Vojislav Bul AVNOJ-a 28 21000 Novi Sad Adamović Dušan 021/366-205

AMSTRAD

Amstradovci — COMPUTER ART vam nudi komplete od 10—15 specijalno izabranih programa po cenl od 600 din + kaseta
Komplet 28. KANE SAI COMBAR KUNG FU MASTER BOMS JACK NIGHT GUNNER komplet 25 GHOST AND GOBLINS, MARSPORT FAIRLIGHT TOMAHAWK komplet 24 SPIHOIZZY PAC MAN GUNFRIGHT GREEN BERET
Sabljak Ivica 7 Vojvođanske brigade 92, 21208 Sremska Kamenica

ŠATANA SOFT nudi Batman, Fairlight, Saboteur Marsport, Samanta Fox i sličnu Katalog besplatan Jovanović Dejan Maršala Tita 48 31330 Priboj, tel 033/53-363

AMSTRAD CPC 464 — Baseball Basketball Gatecracher, Juggernaut, Last V8 Strem, Green Beret Batman Spindizzy Gunfright, Dynamite Dan Fairlight s prelude Shogun Dna man and his dreid Masterbronic Tomahawk, Impossible mission I na kraju Laser Genius (najbolji assembler) Zujić Hinoje, Drvarska 10, 54000 Osijek, tel 064/23-790

Nightmare Soft za CPC 464 Komplet igara Green Beret, Marsport Fairlight, Gun Fright, Starion. Turbo Esprit Movie Batman, Way of Tiger Rambo + kaseta za 2000 din David Leščnik, Maistrova 12, 62000 Maribor

Amstradovci BOMB JACK (pišete Spektrumove), SPACE SHUTTLE (izvrsna simulacija u svemiru) PORNO SHOW-28 digitalizovanih slika TURBO SAVE (skratite učitavanje programa), digitalizator govora (AMSTRAD govori)
niz drugih programa po ceni od 100.— din Super katalog besplatan!
Stojanović Dražen Vojvode Putnika 185 71000 Sarajevo 071/613-349

AMSOFT YU CP/M SOFTWARE predstavlja najnovije CP/M programe DR GRAPH DR PASCAL MT+ STOCKCONTRDL, SUPER DATA INTERCHANGE ZIP (za dBase II), MULTIPLAN, DATASTAR CAMBASE DATABASE TURBO PASCAL, COBOL 80, ALGOL MBASIC mac PRDLOG LISP BASIC-Compiler C-Compiler C-BASIC 80 Compiler ED-1000 FORTRAN 80, WORDMASTER. DISC DOCTOR
Komplet CP/M 2.2 MICROSCRIPT MICROPEN MICRDSPREAD POWER COPYFILE Komplet CP/M 3.0 WORDSTAR 3.34 dBASE II SUPERCALC 2. ZIP COPYFILE Novi CP/M Utility programi TURBO PASCAL TOOLBOX MODULES TURBO SOURCE LISTER SUPERCALC 2 UTILITIES PASCAL MT+ UTILITIES. C-COMPILER UTILITIES
Svaki kupac CP/M programa dobija na pokion CP/M program MINI CADCAM Novi uslužni programi TAW WDRD 128 + Mailmerge PROFI PAINTER DATAMAT TRANSMAT Histoft C-Compiler (na kaseti) DEVPAC 32 (dlec verzije), TURBO DISC (povećava brzinu disca 40%)
Nova literatura C-COMPILER Manual SUPERCALS Manuel CP/M DPERATING SYSTEM MANUAL, CP/M PLUS DPERATING SYSTEM MANUAL C+ PROGRAMMING language MANUAL
Kompleti najnovijih igara (10 do 20 programa) sa kasetom 2000 — din.
AMSOFT YU Trg Republike, 9 41000 Zagreb Telefoni 041/315-470 ili 041/270-777

WAY DF THE TIGER TURBO ESPRIT LAST V8 SHOGUN u kompletu pojedinačne Cena jedneg kompleta (14 + kaseta) — 1600 din Tražite besplatan katalog. Matić Vladan i L. Petara 38 11319 Miloševac

Prodajem AMSTRAD 6128 za zelenom monitorom Modulator M1M2 za bilo televizore Diskete 3" Štampač DMP 2000 Sve je novo i ocarinjeno
Telefon 011/555-785 Beograd (11134, Ul Petra Lubarde br 3

45/mali oglasi

ARCADIA SOFT — i dalje atraktivne nove igre!
komplet 28 KANE PEEP SHOW ZOIDS STORM BARRY Mc GUIGAN BOXING i nešto starija igra GHOST AND GOBLINS GODSHAME MAT FAIR LIGHT GATECRASHER SPELLBOUND cena jednog kompleta 1000 din. Tražite besplatan katalog. Do izlaska Računara još i STARION COL LAPSE TREAD BANGER PETAK 13 CAULDRON 2 Ivanović Milan Nikola Đurkovića 6 11000 Beograd 011/476--423

AMSBYTE DIVISION vam nudi veliki izbor programa za vaš MASTRADSCHNEIDER. Među prako 250 programe možete naći i super hitove MOVIE WAY OF THE TIGER THREE WEEKS IN PARADISE GHOST AND GOBLINS TURBO ESPRIT. Tražite besplatan katalog. Nenad Vasiljević Narodnog fronta 5/II 11000 Beograd tel 011/686 782

BAJA SOFT Veliki izbor programa za AMSTRAD/SCHNEIDER CPC 464 po vrlo prihvatljivim cenama. Uskoro i kod nas INTERNATIDHAL KARATE WEST BANK SDMB JACK profesionalna i brza usluga.
Ujlić Blažo Ante Znanića 15a 88000 Mostar tel 088/415-203

CPC SOFT nudi za vaš AMSTRAD programa na kasetama diskatama kao i literaturu. Među 000 programa koje posedujemo još prošlog meseca bili su ZOIOS KANE STORM MOVIE za programare imamo OCEAN-ov hit LASER GENIUS a za ona sa diskom CP/M programe Dbase II Cobol PRDFI pointer Mini Office 2 igru CAUL DRONE 2 i mnoge druge. Od literatura posedujemo 17 knjiga uputstava ili prevoda s među njima i najnoviju knjigu THE AMAZING AMSTRAD. Naručite besplatan katalog. Mihailo Ivkić Uskočka 7/7 11000 Beograd 011/626-412

Najnovije i najbolje programe CANE GUN FRIGHT STROM JUGGERNAUT MDVIE i druge programe možete jeftino dobiti kod HOT SOFT-a. Narudžbe besplatan katalog
Aksentijević Željko Ženskih heroina Veljka Vlahovića 94 tel 054/45-273 Katić Bernard, Šamačka 3 tel 054/21-018 54000 Osijek

ATARI SOFT-CLUB Zrenjanin 600XL/800XL/130XE programi i literatura. Prodaja, razmena i sređenje izuzetno kursevi za učenje BASIC a na kasetama. Za opširan katalog poštali 100 din. Lacmanović Dejan Sindelićeva 31/e 23000 Zrenjanin tel 023/66-873

Amstradovci BROTHER S SOFT vam nudi veliki izbor starih novih i najnovijih programa po najpovoljnijim cenama. Tražite naš katalog. Besplatan i ilustrovan tako da će vam sve biti jasno. Koldo Otevsal i Nedžad Dinarska 29 71000 Sarajevo tel 071/646-386

ILE SOFT neverovatna !!! 35 šlabranih programe za 1600 din Uputstva za sve programe 500 din Detaljna informacije non-stop na 061/225-523, Vitanov Ilija Skopje Jurija Gagarina 41a

ATARI

ASCII SOFT nudi vam najbolje programe za vaš ATARI 500 XL/130 XE The last V8 Montezuma Revenge Spy Hunter Zorro Zenji James Bond Strip Poker Goonies Hacker Road Race F-15 NOVA literatura i novi programi za disk jedinicu. Upravo je stigao i novi program za učenje bejzika i upoznavanja ATARI-ja (2 kasete C60 + opširne uputstvo) Pandurov Zoran, Đurđevska 43 23000 Zrenjanin 023/63-521

Prodajem-menjam programe jeftino Šaljem besplatan katalog
Sekolov Dragan Karpoševe vostanije 8 A10 1/10 Karpoš 3 91000 Skopje tel 091/258-397

ATARI ST prodajem programe posamezno ali v kompletih. Katalog 400 — din IGY SOFT (062 865 464)

ELEKTRON

Acorn ELECTRON — razmenjujem i prodajem programe (Elite Chuckie Egu Forth Logo Pascal. Besplatan katalog. Adresa Toni Stojanovski Isaija Maževski 91000 D Petrov Skopje tel 091 71 538

GALAKSIJA

ILE SOFTVARE 5 igara telegrafija izradenje 1600 din Zvuk preko kasetofona visoka rezolucija (sve bez hardverskih proširenja). Dosad neviđeni efekti Vitanov Ilije Skopje Junije Gagarina 41a tel 091/225 523

IBM

GTS Elektronika prodaje za 3.950.000 din
Kompletan dual-SCREEN CAD sistem PC XT 640T kolor graf monitor printer 160 chi/sec plotter 120 mm/sek. miš koprocesor programski paket AUTOCAD tel 065/238-036

GTS Elektronika prodaje za 3.950.000 din
PC XT 640T (100% kompatibilan sa IBM) RAM 640K TURBO mode multi I/O kartica TTL graf monitor HERCULES graf kartica 2 flopija po 360 K, HARD 21.5 MB XT tastatura DOS 3.1 + BASIC Dostava puštanje u rad i jedan dan obuke Garancije 1 godina zamena u roku od 24 sata Tel 055/238-026

HARDVER

Prodajem memorijsko proširenje (64K) za GALAKSIJU (15000 din) i komplet za proširenje memorija SPEKTRUMA na 48K III 80K (12000 din) Boris Lovošević Jakšićeva 35 41000 Zagreb

Komodor C64/C128 — kasetni programi po izboru 20—35 disketni 100—250 s CP/M programi 500—800 din) po programu
Konektor za priključnim kablom joystick (1700) kasetofon (1600) USER-port (2400) produžni kabel za joystick (3000) diskete 5,25 OSDD (1300) katalog besplatan isporuka odmah HARDVER/SOFTVER C64/C128 pp 83 74400 Derventa tel 074/832-832 i 833-778

Prodajem potpuno nov i ocarinjen Commodore C-16 sa kasetofonom i 32 programa. Igor Grčić, Nehruova 132/42 tel 011/162-702 N. Beograd

Povoljno prodajem ZX Spectrum 48 K sa postojjem kasetofonom džojstikom Interfejsom raznom literaturom i 55 kaseta sa oko 650 programa. Đakovački Igor, Karlovačka 18 11000 Zemun Tel 011/219-702

Prodajem Commodora 64 flopi disk, matrični štampač i ploter 012/83-479 od 16—20 časova Vladić Dobrivoje Vrabačova 20, 12220 Veliko Gradište

KEMPSTON INTERFACE
Povežite svo) SPECTRUM sa pa ličkim za igranje. Mogućnost priključivanja svih postojećih džojstika Garantovan kvalitet. Cena 7750 din + reset (1000 din) Uverite se! I CHAMP Hardware Jevica Petrović V Karađorđa 46 61300 Kumanovo

Izrađujem EPROM-programator za Commodora 64 te pripadajem odgovarajući listing programa (BASIC + ASSEMBLER-ski listing)
Dolinar Andrej tel 051/731-461

COMPUTER SERVIS
— Sinclair Commodore Amstrad Atari
— brzi i kvalitetni popravci
— ugradnja reseta i druge usluge
dipl. ing Kovačić Anđelko
VIII Vrbik 33a/6
41000 Zagreb
tel (041/539-277 od 10 do 17 sati

Prodajem AMSTRAD CPC 464 za zeleni monitorom zajedno sa 20 brojeva časopisa Moj Mikro i 60 programa na 8 kaseta. Matija Novaković Strosmajerova 40 23000 Zrenjanin tel 023/61-865

KOMODOROVCI !!! Ako želite da preko zvučnika svog televizora čujete muziku iz originalnog komodorovog kasetofona, tada naručite sklop koji se priključuje na C64 Sheme + sklop + program 1500 din Informacija Rajić Stipo Stojana Mešića 33 Slavonska Požega 55300 tel 056/75-921

Prodajem BBC B dvostruki drajv Cumana 400K 32K bočnog RAM-a + 40 diskete za programima i najboljim ROM softverom Literatura, 30 časopisa. Sve zajedno Žarko Supičić, Draškovićeva 27 Zagreb tel 041/417-748

HARD and SOFT
KAKTUS JOY — novi superjoy kvalitetna Izrada garantira precizno upravljanje i dugi vijek trajanja. Jedinstven oblik, posebno pogodan za držanja u rukama. Nešto zaista novo
G-ROM ZA PRINTER MPS 802 omogućuje prenos Hi-res grafike iz Simmon a BASIC-a Newsrooma Doodle itd 6 Grafičkih naredbi, 8 sboova, 10 slobodno programiranjih znakova. Rezolucija 640×400
Poslobnije informacije na adresi HARD and SOFT P P 163 51000 Rijeka

Prodajem univerzalni INTERFACE za ZX Spectrum (O/D izlaz i ulaz, CENTRONIX Interface A/D D/A pretvarač) Cena po dogovoru
Santalija Bernard 92/3 Rabac 52221 tel 052/872-279 i 872-156.

AMSTRADOV izlaz na TV (modulator) i ispravljač prodajem Garancija godinu dana. Leon Kuna Mihanovićeva 18/3 43500 Daruvar tel 043/31-883

SPECTRUM HAROVER
Veliki izbor dodatnih uređaja vrhunska kvaliteta po povoljnim cijenama. Specijalni popust na komplet Kempsten interface + Radiostik palica. Informacija 058/589-887 P N P electronic, Jerotova 12, 58000 Split

COMMODORE HAROVER
Veliki izbor dodatnih uređaja vrhunska kvaliteta po povoljnim cijenama. Specijalni popust na ROM moduls P N P electronic Jerotova 12, 58000 Split tel 058/589-887

ATARI ST HAROVER
Veliki izbor dodatnih uređaja vrhunska kvaliteta po povoljnim cijenama. RAM 1 Mb ROM-ovi TV modulator disk 720 Kb Fast Basic kartridge literatura programi Telefon 058/589-887 P N P electronic, Jerotova 12 58000 Split

Literatura

KOMODOR C64/C128 — prevodi Priručnici C64 (600) C128 (1550) Simon s BASIC (600) Easy Script (300) Vizawrite (350) Pascal (300) Help C64 (300) Mastr 64 (400) Praktikalk (650) Monitor (200) CP/M c64 (300) Wordstar CP/M C64 (800) COBOL CP/M C64 (500)
Katalog besplatan isporuka domah.
HARDVER/SOFTVER C64/C128 PP 83 74400 Derventa tel 074/832-832 i 833-775

Spectrum Rečnik Englesko—Srpskohrvatski s objašnjenjima (oko 2000 najčešće upotrebljavanih reči) + kaseta + poštarina (1500 din) tel 011/497-862 od 17—19 h O Marjanović B Jerkovića 123/36 — 11000 Beograd

COMMODORE 64 — profesionalni prevodi
Programmer s reference guida — 3800 d Mašinski jezik za početnike — 2400 d Grafika i Zvuk — 2400 d basic Priručnik — 2400 d Simon s Basic — 2400 d Pascal — 1200 d
Na višestruke narudžbe popust 10%
Dutko Bjelotomić, Centar 1 54550 Velpovo
tel 054/82-865 Hi 041/883-141

Spectrum rečnik kompjuterskih izraza i reči upotrebljavanih u kompjutersko tehnici + kaseta + poštarina — 1400 din. tel 011/497-862 od 17—19 h.

AMSTRAD CPC 464 — profesionalni prevodi
Priručnik — 3000 d Locomotive BASIC — 2600 d Mašinsko programiranje za početnike — 2600 d
Višestruka narudžba 10% popust.
Dutko Bjelotomić Centar 1 54550 Velpovo tel 054/82-865 hi 041/883 141

HEACKERJI KNJIGE ZA VAS C Programming Language (2000) C Guide (1300) Logo Programming (1600) Forth Programming (2500), Pascal User Manual (2000) Forthran — 700 stranica (4000) i M68020 — bočunost (4000). TINE JARM ŠENTLOVRENC 20 66212 VELIKA LOKA

crtanje na mašincu (2)

Svrha PLOT rutina je da na ekranu nacrta tačku. Rutina može biti konačnog i neodređenog tipa.

Amstradova PLOT rutina iz ROMe ne zadovoljava karakteristike kvaliteta koje treba da ima PLOT rutine dobrog grafičkog programa i to zbog sporosti iscrtavanja tačaka. To je zbog toga što je ta rutina predviđena za red u sva tri mode (grafički modovi MODE 1, MODE 2 i MODE 0) i zato što je predviđena da crta tačku u izlomljenoj bitnoj mapi (poseb hardverskog SCROLLa). Pošto se u grafičkim programima na upotrebljavaju sva tri mode odjednom, niti se upotrebljava hardverski SCROLL, možemo znatno povećati brzinu rutina odričući se pomenutih svojstava

Izvan uhodanih staza

PLOT rutina može da bude formirana na dva načina koji imaju zadovoljavajuću brzi-

```
Faza 1 arrora: 00

5200                    1d          org  21000
5200                    20          ast  d
5200  F3                00          di
5201  0707              40 plot:    ld   a,7
5203  47                00          ld   b,a
5204  79                00          ld   a,c
5205  A1                78          and  c
5206  E644              00          sag
5208  81                00          add  c,c
5209  87                100         add  a,a
520A  87                110         add  a,a
520B  87                120         add  a,a
520C  FE08              130         or   %11000110
520E  3E5E52            140         ld   (hb),a
5211  CB3b              100         ari  d
5213  CB1b              100         rr   e
5215  CB3a              170         ari  d
5217  CB1b              100         rr   e
5219  CB3A              100         ari  d
521B  CB1b              350         rr   e
521D  83C8              210         ld   a,(hb)
5221  82                220         add  a,d
5222  67                300         ld   h,a
5223  6E                310         ld   l,(hl)
5224  2B                320         dec  hl
5225  7B                330         ld   a,e
5226  87                340         add  a,a
5227  87                350         add  a,a
5228  87                360         add  a,a
5229  87                370         add  a,a
522A  87                400         add  a,a
522B  87                410         add  a,a
522C  87                420         add  a,a
522D  87                430         add  a,a
522E  87                440         add  a,a
522F  87                450         add  a,a
5230  87                460         add  a,a
5231  87                470         add  a,a
5232  87                500         add  a,a
5233  87                510         add  a,a
5234  87                520         add  a,a
5235  FE08              530         or   %11000110
5237  32F7E2            540         ld   (hb),a
523A  CB3A              550         ari  d
523C  CB1b              560         rr   e
523E  CB3D              570         ari  d
5240  CB1b              580         rr   e
5242  CB3A              590         ari  d
5244  CB1b              600         rr   e
5246  83C8              610         ld   a,(52
5248  82                620         add  a,d
5249  57                630         ld   d,a
524A  7E                640         ld   a,(h
524B  8078              650         and  248
524D  87                660         ld   l,a
524E  D0                670         add  h,(h
5250  A1                680         and  c
5251  E3                690         ex   de,(h
5252  18                700         add  h),de
5253  B3                710         or   e
5254  E3                720         ex   de,(h
5255  00                730         add  h),(h
5256  18                740         add  h),de
5257  B0                750         or   h
5258  A1                760         and  c
5259  87                770         add  a,a
525A  87                780         add  a,a
525B  87                790         add  a,a
525C  84                800         add  a,h
525D  67                810         ld   h,a
525E  CB00              820 hb:     rlc  (hl)
5260  C9                830         ret

Faza 2 arrora: 00

Table used:     42 free    175
Executed: 21000
```

Primer 1
Ulazni parametri A — kreće se od 0 do 199 DE — kreće se od 0 do 639 zaprijeni su evi registri osim i registre


```
Faza 1 arrora: 00

5200                    10          org  21000
5200                    20          ast  0
5200  CB14BC            00          call 0be14
5203  F3                40          di
5204  1177F82           00          ld   de,639
5207  3EC7              60          ld   a,199
5209  FD47              70          ld   i,a
520B  D5                00 st:      push de
520C  3B57              00          ld   d,i
520E  CD2A52            100         call plot
5211  E1                110         pop  de
5212  7A                120         ld   a,d
5213  B3                130         or   e
5214  1B                140         dec  de
5215  2074              100         jr   nz,st
5217  FD57              100         ld   a,i
5219  C9                170         ret  h
521A  3D                200         dec  a
521B  E5444             200         sbg
521E  A1                200         and  c
521F  47                270         ld   b,a
5220  87                300         add  a,a
5221  87                310         add  a,a
5222  87                320         add  a,a
5223  87                330         add  a,a
5224  87                340         add  a,a
5225  FE08              350         or   %11000110
5227  32F7E2            320         ld   (hb),a
522A  CB3A              330         ari  d
522C  CB1b              300         rr   e
522E  CB3D              300         ari  d
5230  CB1b              300         rr   e
5232  83C8              300         ld   a,(52
5234  82                400         add  a,d
5235  57                410         ld   d,a
5236  7E                420         ld   a,h
5237  8078              430         and  248
5239  2808              440         jr   h e
523B  67                450         ld   l,a
523C  E3                460         ex   de,hl
523D  19                470         add  hl,de
523E  E3                480         ex   de,hl
523F  00                490         nop
5240  A1                500         and  c
5241  87                510         add  a,a
5242  87                520         add  a,a
5243  87                530         add  a,a
5244  87                540         add  a,a
5245  87                550         add  a,a
5246  87                560         add  a,a
5247  87                570         add  a,a
5248  84                580         add  a,h
5249  67                590         ld   h,a
524A  CB00              600 hb:     rlc  (hl)
524C  C9                610         ret

Faza 2 arrora: 00

Table used:     42 free    175
Executed: 21000
```

Primer 2
Nema ulaznih parametara.

nu. Jedan od načina je rutine koje koristi mašinske instrukcije SET, RES i BIT, a drugi način je formiranje rutine korišćenjem mašinskih naredbi AND, OR i XOR

Svaki od ovih načina ima svoje kvalitete i svoje mane. Prvi tip može istovremeno da redi i kao TEST i kao UNPLOT rutina, a ne može da redi kao INVERT rutina. Drugi tip PLOT rutina ne može da redi kao TEST, a redi kao INVERT rutine, tj. može da invertuje tačku. Prvi način je nešto brži od drugog

Ulazni parametri PLOT rutina „emstradovog" ROMe upisuju se u DE i HL registre, pri čemu se DE kreće od 0 do 639 a HL od 0 do 398. Ovde DE označava tačke na X osi, a HL tačke na Y osi. Ova rutina tretira svaku drugu liniju, što je razumljivo jer „amstrad" ime samo 200 linija. Ulazni parametar rutina koji označava broj reda može se staviti u osmobitni A registar i pri tom dobiti ne brzini i jednostavnosti rutine. Tada se redni broj linija kreće od 0 do 199

Prvo što PLOT rutina treba da preračuna od rednog broja deta tačke je da izračuna adresu bajta u kome je tačka, a zatim broj bita koji treba da se setuje u bajtu. Pošto svaki bajt ima 8 bita, broj bajta u redu se dobije tako što se redni broj tačke podeli se 8 (Primer 1: linijski brojevi od 150 do 200). U binarnom sistemu 8 je ceo broj (8=1000).

Pozneto je da se šiftovanjem bajta za jedno mesto udesno vrši deljenje se 2. Ako se tri puta uzastopno izvrši šiftovanje sadržaj bajta se deli se 8 ($2^3=8$). Tako se deljenje broja se 8 može izvršiti se tri mašinske naredbe *SRL reg*

Broj bita u bajtu se dobija kao ostatak pri deljenju rednog broja tačke se 8. Osta

```
Faza 1 arrora: 00

5200                    10          org  21000
5200                    20          ast  0
5200  F3                00          di
5201  47                40 plot:    ld   b,a
5202  4A                00          ld   c,a
5203  CB3A              00          ari  d
6300  CB1b              70          rr   e
5207  CB3A              00          ari  d
5211  CB1b              00          rr   e
5213  CB3A              100         ari  d
5215  CB1b              110         rr   e
5217  1B2B              120         ld   a (52
5218  82                130         add  a,d
521A  57                140         ld   d,a
521B  70                100         ld   a,h
521C  E678              100         and  248
521E  2808              170         ld   h,0
5220  6F                100         ld   l a
5221  2B                100         add  hl,hl
5222  1A                360         ex   de,hl
5223  18                210         add  hl,de
5224  E3                220         ex   de,hl
5225  00                230         add  hl,hl
5226  29                240         add  hl,hl
5227  18                250         add  hl,de
5228  B0                800         or   h
5229  E607              270         and  7
522B  87                800         add  a a
522C  87                290         add  a a
522D  87                300         add  a,a
522E  84                310         add  d h
522F  67                320         ld   h,a
5230  78                330         ld   a,c
5231  E607              340         and  7
5233  47                350         ld   b a
5234  3E80              360         ld   a 128
5236  07                370 aa:     rrca
5237  10FA              360         djnz aa
5239  80                360         or   (h
523A  77                400         ld   (hl) a
523B  C9                410         ret

Faza 2 arrora: 00

Table used:     33 free    140
Executed: 21000
```

Primer 3
Vidi primer 1

```
Pass 1 errors: 00

5200                    10          org  21000
5200                    20          ent  0
5200  F3                30          di
5201  43                40 start:   ld   b,e
5202  CB04              50          srl  d
5204  CB1B              60          rr   e
5206  CB3A              70          srl  d
5208  CB1C              80          rr   e
520A  CB0A              90          srl  d
520C  CB1C              100         rr   e
520E  21FB52            110         ld   hl,(bp)
5211  18                120         add  hl,de
5212  78                130         ld   a,b
5213  B0E7              140         and  7
5215  47                150         ld   b,a
5216  3E80              160         ld   a,128
5218  07                170 bit:    rrca
5219  10FD              180         djnz bit
521B  B6                190         or   (hl)
521C  77                200         ld   (hl),a
521D  C9                210         ret
521E  00C0              220 bp:     defw 49152

Pass 2 errors: 00

Table used:     42  free    128
Executes: 21000
```

Primer 4
Ulazni parametri: DE — kreće se od 0 do 639. lokacija na adresi HP sadrži adresu reda u koma se crta tačka. Nepromanjani su registri C i I.

tak se može izračunati ako se nad lakšim bajtom rednog broja tačka izvrši logične operacija AND 7 (Naiaženo ostatka pri deljenju rednog broja se 8 opisano ja u prethodnom broju „Računara") Linija 70 primera 1.

SET, BIT i RES

SET, RES i BIT naredbe spadaju u dvobajtne naredbe procesora Z80 Od značaja za ovu rutinu su samo naredba koja se izvršavaju nad bajtom na adresi HL BIT b, (HL) SET b, (HL) i RES b, (HL)

Prvi bajt ovih naredbi je CB, a drugi bajt je diraktno zavistan od ulaznih parametara

```
Pass 1 errors: 00

5200                    10          org  21000
5200                    20          ent  b
5200  F3                30          di
5201  78                40 start:   ld   a,e
5202  B007              50          and  7
5204  B04b              60          sug
5205  C607              70          add  a,7
5207  87                80          adb  a a
5208  87                90          abb  a a
5209  87                100         add  a,a
520A  FECB              110         or   0110001(0
520C  3E2852            120         ld   (hbell),a
520F  C33A              130         srl  d
5211  CB1B              140         rr   e
5213  C33A              150         srl  d
5215  CB1B              160         rr   e
5217  C38A              170         srl  d
5219  CB1B              180         rr   e
521B  2A2b52            190         ld   hl,(bp)
521E  19                200         add  hl,de
521F  CB40              210 bit:    bit  0,(hll
5221  C9                220         ret
5222  00C0              230 bp:     defw 49152

Pass 2 errors: 00

Table used:     42  free    128
Executes: 21000
```

Primer 5
Ulazni parametri: D-kreće se od 0 do 639, lokacija na adresi HP sadrži adresu reda u kome se crta tačka. Registar BC je nepromenjen


```
Pass 1 errors: 00

5200                    10          org  21000
5200                    20          ent  0
5200  F3                30 begin:   di
5201  AF                40          xor  a
5202  0600              50          ld   b,a
5204  1160C0            60 st:      ld   de,49152
5207  21A001            70          ld   hl,40000
520A  00E               80 ett:     rrc  (hll
520C  23                90          ex   de,hl
520D  CB1E              100         rr   (hl)
520F  EB                110         ex   de,hl
5210  13                120         inc  de
5211  22                130         ioh  hl
5212  0A                140         cp   d
5213  20F0              150         jr   nz,stt
5215  10E5              160         djnz st
5217  C9                170         ret

Pass 2 errors: 00

Table used:     43  free    122
Executes: 21000
```

Primer 6
Nema ulaznih parametara

Zadnja tri bita drugog bajta naredbe su uvek lata I iznosa 1, 1, 0 Prva dva bita određuju da li se radi o komandi BIT SET ili RES Ako su upaljena oba bita (1,1), radi se o komandi SET Ako je upaljen prvi bit (1, 0) radi se o komandi RES, a ako je upaljen drugi bit (0,1) radi se o komandi BIT Preostali bitovi određuju broj bita u bajtu na adresi HL koji će biti tratiran Adresa bajta u VIDEO memoriji se dobija tako što se redni broj bajta u redu sabara se adresom rada Adresu reda izračunava modifikovane A HL rutina u okiopu sema PLOT rutine (Primar 1 linijaki brojevi 210 do 410, brzina PLOT rutina se može videti ako se u računar unese primer 2)

Rutina se može prapraviti u SET ili RESET rutinu promenom bita u broju iza OR u linlji 130 (Primer 1)

AND, OR i XOR

PLOT rutina koje koristi AND, OR i XOR mašinske naredba je nešto sporija od

```
Pass 1 errors: 00

0200                    10          org  21000
0200                    20          ent  0
0200  018A7F            30 prinst:  ld   bc,07f0a
0203  BD49              40          ean  (hl),c
0205  2800              50          ld   h,0
0207  07                60          ld   l,a
0208  0E30              70          ld   c,50
020A  20                80          add  hl,hl
020B  20                90          add  hl,hl
020C  20                100         add  hl,hl
020D  84                110         add  a,h
020E  67                120         ld   h,a
020F  09                130         ex   de,hl
0210  2A3052            140         ld   (hp),hl
0213  EB                150         ex   de,hl
0214  223052            160         ld   (hp),hl
0217  018808            170         ld   bc,2000
021A  1A                180 st:     ld   a,(hl)
021B  77                190         ld   (hll a
021C  23                200         ioc  de
021D  13                210         ld   a,c
021E  04                220         add  a,b
021F  07                230         ld   b a
0220  1078              240         djnz st
0222  018077            250         ld   bc,07f0a
0225  EDb0              060         ent  ici,c
0227  C9                270         ret
0228  FFAF              000 bp:     defw 49101

Pass 2 errors: 00

Table used:     43  free    130
Executes: 21000
```

Primer 7
Ulazni parametri: A-sadrži ASCIi kod slova koje se lapisuje Svi registri osim registra, i su zaprijani Napomena Rutina mora biti iznad 16384 bajta memorijel Obavezno onemogućiti interapt pra poziva rutinal

```
Pass 1 errors: 00

5200                    10          org  21000
5200                    20          ent  0
5200  CB14BC            30          call 0bold
5203  F3                40          di
5204  21FF07            50          lh   hl,40151
5207  224852            60          ld   (hp) hl
520A  018007            70          ld   bc,2000
520D  C5                80 st:      push bc
520E  79                90          ld   a,c
520F  CB2152            100         call priat
5212  C1                110         pop  bc
5213  00                120         dec  bc
5214  78                130         ld   a,b
5215  01                140         or   c
5216  20F5              150         jr   nz,stt
5218  C9                160         ret
5219  018A7F            170 priat:  ld   bc,07f0a
521C  ED40              180         ent  ici,c
521E  2600              190         ld   h,0
5220  07                200         ld   l,a
5221  3E30              210         ld   a,50
5223  29                220         add  hl,hl
5224  29                230         add  hl,hl
5225  29                240         add  hl,hl
5226  84                250         add  a,h
5227  67                260         ld   h,a
5228  EB                270         ex   de hl
5229  2A4852            080         ld   hl,(hp)
522C  23                290         inc  hl
522D  224852            300         ld   (hp),hl
5230  018000            310         ld   bc,2050
5233  1A                000 st:     ld   a,(de)
5234  77                000         ld   (hll,a
5235  13                340         ina  de
5236  78                350         ld   a,c
5237  84                300         add  a,h
5238  67                370         ld   h,a
5239  10F3              000         djnz st
523B  018077            390         ld   bc,07f0a
523E  ED49              400         ent  ici,c
5240  C9                410         ret
5241  FFAF              420 bp:     defw 49101

Pass 2 errors: 00

Table used:     53  free    150
Executes: 21000
```

Primer 8
Nema ulaznih parametara

prathodne rutina all na menja semu seba, pa se može koristiti u ROMu Rutina se razlikuje od prethodna semo u načinu setovanja bita u bajtu Pošto rutina izračunava poziciju bita u bajtu, ona taj broj prebacuje u B registar (Primar 3 linija 350), a u A registar upisuja broj 128, tj u binarnoj formi broj &x10000000

Svaki put kada se obrne petlja A registar se rotira za 1 bit udseno Posle izvršena petlja broj setovanih bita u A registru jednak je broju bita u bajtu koji bi imala data tačka. Posle toga se A registar ORuje sa bajtom na adresi HL (Primer 3 linija 390) Time je izvršeno crtanje tačke Ako se više PLOT rutina izvršava u latoj liniji (npr u horizontalnoj kompresiji) nije potrebno da se svaki put izračunava adresa reda, već sa adresa reda može dobiti pomoću AHL rutina I staviti ne određenu lokaciju, na početku procesa odakle je zatim uzima PLOT rutina Takve PLOT rutina je znatno brža i kraće od PLOT rutine koja sadrži modifikovanu AHL rutinu u avom sastavu (Primar 4 i 5) Rutina se prebacuje u INVERT mod promenom linije ,390 or (hi) u „390 xor (Hl)

Prikazivanje uputstva

Prilikom pisanja grafičkih programa javlje se problam prikazivanja uputstava.

Uputstvo se može čuvati u memoriji na dva načina u vidu slika I u vidu takata.

Čuvanja manije u vidu slika je vrlo jednostavan način, ali zauzima 16 kilobajta

```
10 'Vertikalni kompresor
20 FOR xx=0 TO 638:FOR yx=0 TO 398 STEP 2:IF TEST(xx,yx) THEN PLOT xx,yx,0:PLOT
xx,yx%2,13
30 NEXT:NEXT


10 'Horizontalni kompresor
20 FOR yx=0 TO 398 STEP 2:FOR xx=0 TO 638:IF TEST(xx,yx) THEN PLOT xx,yx,0:PLOT
xx/2,yx,13
30 NEXT:NEXT
```

```
Pass 1 errors: 00

5200                    10          org  21000
5200                    20          ent  0
5200  F3                30          di
5200  3EC7              40 comp:    ld   a,199
5203  0600              50          ld   b,0
5205  CD2952            60 ati:     call A_HL
5208  B347              70          ld   l,a
5210  E5                80          push hl
5211  CB5F              90          srl  a
5213  CB84              100         add  a,100
5215  CD2952            110         call A_HL
5218  D1                120         pop  de
5219  4B                130         ld   c,b
521C  0650              140         ld   b,80
521E  1B                150         ex   de,hl
521F  1A                160 et:     ld   a,(de)
5220  B0                170         or   (hl)
5221  12                180         ld   (de),a
5222  71                190         ld   (hl),c
5223  23                200         inc  hl
5224  13                210         inc  de
5225  10FB              220         djnz et
5227  3D57              230         ld   a,s
5229  C8                240         ret  z
522A  38                250         dec  a
522B  18E9              260         jr   ati
522D  11400C            270 A_HL:   ld   de,49152
5230  47                280         ld   c,a
5231  E6F8              290         and  248
5233  63                300         ld   h,e
5234  6F                310         ld   l,a
5235  29                320         add  hl,hl
5236  29                330         ex   de,hl
5237  19                340         add  hl,de
5238  EB                350         ex   de,hl
5239  29                360         add  hl,hl
523A  29                370         add  hl,hl
523B  19                380         add  hl,de
523C  51                390         or   c
523D  E607              400         and  7
523F  B7                410         add  a,a
5240  B7                420         add  a,a
5241  B7                430         add  a,a
5242  B4                440         add  a,h
5243  B7                450         add  a,a
5244  78                460         ld   a,c
5245  C9                470         ret

Pass 2 errors: 00

Table used:     54 free    186
Executes: 21000
```

Primer 10
Nema ulaznih parametara

slobodne memorije. U grafičkim programima se primenjuju dva načina takvog čuvanja slika. Prvi način je izmena sadržaja memorija i video memorije a drugi način je da se u memoriji formiraju dva video memorija (ukupno 32 kilobajta). Zatim, pre ispisivanja uputstava, slika traba staviti u drugu video mamoriju, a uputstvo iz prve lokacija prebaciti u video mamoriju. Tej način oduzima 32 kilobajta, tako de ostaje slobodno samo 10 kilobajta, što je dovoljno za grafički program.

Vrlo interesantni efekti ispisivanja uputstva mogu se postići primanom programa iz primara 6. Liniju 70 ovog programa traba izmeniti tako da umesto 25000 stoji adresa menija. Ovo se može postići spuštanjam VIDEO mamorija u prvu lokaciju. Tako se postiže trenuten prelaz se SCREENa na MENI.

```
Pass 1 errors: 00

5200                    10          org  21000
5200                    20          ent  0
5200  F3                30          di
5200  0602              40 vanag:   ld   b,200
5203  78                50 ati:     ld   a,b
5204  35                60          dec  a
5205  CD4752            70          call A_HL
5208  228582            80          ld   (hpl),hl
520B  110000            90          ld   de,0
520E  D5                100 ati1:   push de
520F  0648              110         ld   c,36[5861]10
5211  CD4252            120         call plot
5214  2B1E              130         jr   z,etZ
5216  ED67              140         ld   a,i
5218  F608              150         or   %[0000]10
521A  2A8582            160         ld   [hb+1],a
521D  CD4252            170         call hb
5220  E1                180         pop  de
5221  D5                190         push de
5222  CB3A              200         srl  d
5224  CB1B              210         rr   e
5226  CD4252            220         call plot
5229  E1                230         pop  de
522A  13                240 et2:    inc  de
522B  7A                250         ld   a,d
522C  FE02              260         cp   %2
522E  20DC              270         jr   nz,ati
5230  7B                280         ld   a,e
5231  FE00              290         cp   %00
5233  20D7              300         jr   nz,ati
5235  1600A             310         djnz at
5237  C9                320         ret
5238  78                330 plot:   ld   a,e
5239  E607              340         and  7
523B  47                350         ma   a
523C  7A                360         ld   a,d
523D  E607              370         and  a 7
523F  B7                380         add  a,a
5240  B7                390         add  a,a
5241  B7                400         add  a,a
5242  B7                410         add  a,a
5243  B4                420         add  a,[hb+1] a
5245  CB3A              430         srl  d
5247  CB1B              440         rr   e
5249  CB3A              450         srl  d
524B  CB1B              460         rr   e
524D  CB3A              470         srl  d
524F  CB1B              480         rr   e
5251  2485E2            490         ld   hl,[hpl
5254  18                500         add  hl,de
5255  2048              510 hb:     bit  0 (hl)
5257  C8                520         ret
5258  0E08              530 hpl     data 49152
525A  11400B            540 A_HL:   ld   de,49152
525D  47                550         ld   c,a
525E  E6F8              560         and  248
5260  63                570         ld   h,a
5261  6F                580         ld   l,a
5262  29                590         add  hl,hl
5263  29                600         add  hl,hl
5264  19                610         ma   de,hl
5265  29                620         add  hl,de
5266  29                630         ex   de hl
5267  19                640         add  hl,hl
5268  29                650         add  hl,hl
5269  19                660         add  hl de
526A  51                670         or   c
526B  E607              680         and  7
526D  B7                690         add  a a
526E  B7                700         add  a a
526F  B7                710         add  a,a
5270  B4                720         add  a,h
5271  67                730         ld   h,a
5272  79                740         ld   a e
5273  C9                750         ret

Pass 2 errors: 00

Table used:     54 free    186
Executes: 21000
```

Primer 11
Nema ulaznih parametara

Pošto u Amstradovom JUMP bloku postoji rutina koja to redi, ovda ta mogućnost neće biti razmatrana.

Drugi način čuvanja manlja je čuvanje u obliku takata. Tada mani zauzima samo 2000 bajtova a u video memoriju se ispisuja PRINT rutinom Amstradova. PRINT rutina ja spora (jer je predviđena za rad u sva tri grafička moda i u režimu izlomljene

```
Pass 1 errors: 00

5200                    10          org  21000
5200                    20          ent  0
5200  F3                30          di
5201  3E00              40 A_HL:    ld   a,0
5203  78                50          ld   a,b
5204  3D                60          dec  a
5205  C24752            70          jp   nz A_HL
5208  C9                80          ret
5209  3EF8              90          and  248
520B  67                100         ld   h,a
520C  6F                110         ld   l,a
520D  29                120         add  hl,hl
520E  29                130         add  hl,hl
520F  19                140         add  hl,de
5210  29                150         add  hl,hl
5211  29                160         add  hl,hl
5212  19                170         add  hl,de
5213  C9                180         ret

Pass 2 errors: 00

Table used:     24 free    196
Executes: 21000
```

Primer 12
Ulazni parametri A-kraće se od 0 do 199
Izlazni parametri HL — sadrži adresu reda, A i B su nepromenjeni

bitne mape video memorije), pa je zato napogodna za tu primenu. Ako se odreknemo tih svojstava, možemo napisati PRINT rutinu koja svojom brzinom zadovoljava tražene uslove.

Definicija ASCII koda „amstradovom ROMu počinje od adrese #3800. Ove rutine se mora nalaziti iznad 16384 bajta, jer se u toku rada otvara donji ROM (programske linija 50 i 260).

Brzina ova rutina može se proveriti učitavanjam i startovanjem primara 8. HP bajt sadrži adresu slova na akranu i njegov sadržaj se izračunava po formuli

HP=49151+x+(y-1)*80

x — horizontalna koordinata slova
y — vertikalne koordinata slova

Kompresori slike

Kompresori imaju ulogu vizualnog skupljanja akrane. Ovda ćemo razmotrit kompresora koji vrše horizontalno i vertikalno skupljanje sadržaja akrana svodeći ga na polovinu. Horizontalna kompresija je jednostavnije od vertikalna, jer se može vršiti i nad linijeme a na nad tačkama kao kod vertikalne kompresija.

Bejzik primari kompresija deti su u programu 9. Mašinski primeri programa koja vrše kompraslju dati su u primaru 10 primaru 11. Primer 11 koristi modifikovanu PLOT rutinu u procesu kompresija.

Ovaj put vam nudimo novu AHL rutinu koja je znatno brža i kraća od AHL rutine objavljena u prethodnom broju (Primer 12). Ova rutina ostavlja slobodan B registar tako da se može koristiti u petljams formiranim se DJNZ.

Date rutine zadovoljaveju ave potrebne uslove za primanu u grafičkim programima visoka katagorija.

Andrija Radović

kule bez rekurzije

Pretpostavljamo da ste pročitali prethodni članak o simulaciji rekurzije u „Računarima 15" već prostudirali pa ovog puta bez mnogo filozofiranja prelazimo na zadatak o Hanojskim kulama.

Neka su data tri štapa i n diskova različitog prečnika. Diskovi se mogu slagati na štap samo tako da obrazuju kulu (dakle, donji disk ja najveći, naredni nešto manji itd. i gornji najmanji). Neka sa na početku svi diskovi nalaza na prvom štapu. Zadatak ja napisati program za simuliranje premeštanja svih diskova sa prvog štapa na treći, tako da na kraju svi diskovi budu poređani u istom redosledu koji su imali na početku. Pri premeštanju diskova pridržavati se sledećih pravila:
(1) samo jedan disk se može prenati sa jednog na drugi štap u jednom potezu
(2) veći disk sa na ama staviti preko manjeg diska
(3) drugi štap as može koristiti kao pomoćni

Uz pomoć rekurzije zadatak o Hanojskim kulama rešava se vrlo jednostavno. Pretpostavimo da svaku kulu sa manje od n diskova već znamo da prabacujemo (to ja rakurzivni korak) ideja ja da „skraćenu kulu od n-1 diskova (svi osim najvećeg) prebacimo sa prvog diska na pomoćni disk. Zatim preostali najveći disk prabacimo na ciljni disk. Na kraju prabacimo našu „skraćenu kulu sa pomoćnog na ciljni disk. Rakurzivna procedura koja ovo redi data ja na slici 1.

Pokušajte da bez daljeg čitanja teksta zadatak rešita na korisneći rekurziju. Nije ni malo jednostavno, zar ne?
U dirakinom pokušaju da iterativno rašimo ovaj zadatak najviše glavobolja zadaja nam to što na vidimo jasno naki algoritam prebacivanja koji vodi ka ciju. Zato ćemo sa posluživti atarim lukavatvom. Simuliraćemo rekurziju iz procedure na slici 1 a zatim pokušati da dobiveno iterativno rašenja još malo pojednostavimo.

Procedura sa slika 1 ima četiri parametra, od kojih se svaki može promaniti u rekurzivnom pozivu. Zato oblast podataka mora sadržavati alamanta koji predatavljaju ava četiri. Lokalnih promanljivih nama. Postoji i jedna privremena promanljiva koja čuva vrednost od n-1 ali nama potrebe, da je čuvamo u oblasti podataka. Postoje tri tačke mogućeg povratka iz rekurzivnog poziva pozivni program i dve tačka u programu iza rekurzivnih poziva.

Dakla trebaća nam tri labela

```
labal 1  2, 3
```

Adrasa povratka biće broj (1 2 ili 3) unutar svaka oblasti podataka.

Nakon ovih kraćih razmatranja već smo dovoljno spremni da napišamo iterativnu simulaciju procadure sa slika 1. Ovaj diraktni iterativni ana logon dat ja na slici 2.

Pogadata već — nećemo se zadovoljiti ovim rešenjem istina ono jesta narekurzivno, ali nije previše alagantno. Ima tu još doata alamenata za pojednostavljivanja.

Kao prvo primatimo da smo koristili tri adresa povratka po jednu za svaki od dva rakurzivna poziva i jednu za povratak u glavni program. Međutim povratak u glavni program može sa dataktovati i na drugi način — kao underflow u staku, beš kao u drugoj varziji programa faktorijel iz „Računara 15. Oslaju nam još dva labale povratka. Kada bismo uspeli da alimenišemo još jednu labelu viša na bismo morali da pamtimo adresu povratka na ataku. Pozabevimo se drugim rakurzivnim pozivom.

```
PROCEDURE towers (n: Posint; frompeg,topeg,auxpeg: char);
BEGIN
  IF n=1
    THEN writeln('Prebaci Disk 1 sa stapa',frompeg,'na stap',topeg)
    ELSE
      BEGIN
        towers(n-1,frompeg,auxpeg,topeg);
        writeln('Prebaci Disk',n,'sa stapa',frompeg,'na stap',topeg);
        towers(n-1,auxpeg,topeg,frompeg)
      END (* else begin *)
END (* towers *);
```

slika 1

```
PROCEDURE simtowers (n: Posint; frompeg, topeg, auxpeg: char);
LABEL 1,2,3,10;
CONST maxstack=50;
TYPE  Dataarea= RECORD
                  nParam: Posint;
                  fromParam: char;
                  toParam  char;
                  auxParam: char;
                  retar: 1..3
                END;
      stack= RECORD
               top  0..maxstack;
               item: array[1..maxstack] of Dataarea
             END;
VAR s: stack;
    currarea: Dataarea;
    i: 1..3;
begin
    (* inicijalizacija *)
    s.top:=0;
    currarea.nParam:=1;
    currarea.fromParam:=' ';
    currarea.toParam:=' ';
    currarea.auxParam:=' ';
    currarea.retar:=1;
    (* Pakovanje iazmeo elementa ni stek *)
    Push(s,currarea);
    (* setevanje parametara i adrese povratka tekuce
       oblasti Podataka na odgovarajuce vrenosti *)
    currarea.nParam :=n;
    currarea fromParam :=frompeg;
    currarea toParam:=topeg)          a
    currarea.auxParam:=auxpeg;
    currarea.retar :=1;
10  (* Pocetak simulirane rutine *)
    iF currarea.nParam=1
     THEN
      WITH currarea DO
       BEGIN
        writeln('prenesi  Disk  1  sa  stapa',fromParam
                'na stap',toParam);
        i :=retar;
        Popsub(s,currarea);
        CASE i OF
          1: GOTO 1;
          2: GOTO 2;
          3: GOTO 3
        END (* case *)
       END (* then with *);
    (* Prvi rekurzivni Poziv *)
    Push(s,currarea);
    currarea.nParam:=s.item[s.top].nParam-1;
    currarea.fromParam:=s.item[s.top].fromParam;
    currarea.toParam:=s.item[s.top].auxParam;
    currarea.auxParam:=s.item[s.top].toParam;
    currarea.retar:=2;
    GOTO 10;
2:  (* tacka Povratka iz Prvog rekurzivnog Poziva *)
    writeln('Prenesi Disk', currarea.nParam,'sa stapa',
              currarea.fromParam,'na stap',currarea.toParam);
    (* Drugi rekurzivni Poziv *)
    Push(s,currarea);
    currarea.nParam:=s.item[s.top].nParam-1;
    currarea.fromParam:=s.item[s.top].auxParam;
    currarea.toParam =s.item[s.top].toParam;
    currarea.auxParam:=s.item[s.top].fromParam;
    currarea retar:=3;
    GOTO 10;
3   (* tacka Povratka iz drugog rekurzivnog Poziva *)
    i:=currarea.retar;
    Popsub(s,currarea);
    CASE i OF
      1: GOTO 1;
      2: GOTO 2;
      3  GOTO 3
    END (* case *)
1:  (* Povratak u glavni Program *)
END (* simtowers *);
```

slika 2

towers (n-1 auxpeg topag frompeg)
and

Ovaj poziv simuliramo kao
(1) Pakujamo tekuću oblast podataka a1 na atek

(2) Postavljamo parametra u novoj tekućoj oblasti podataka a2 na odgovarajuće vrednosti n-1 auxpeg topeg frompeg
(3) Postavljamo adresu povratka u tekućoj oblasti podataka a2 na adrsau naredbe odmah iza poziva

U Računarima 15 objasnili smo kako simulirati tehniku rekurzije. Priču smo potkrepili školskim primerom (funkcija faktorijel) koji ste sigurno i ranije, bez našeg mudrovanja, mogli lako da napišete u iterativnom obliku.
Ovog puta simulacijom ćemo rešiti nerekuzivno zadatak čije direktno iterativno rešavanje ne ide tako glatko.

```
PROCEDURE s
CONST maxstack=50;
TYPE  Dataarea= RECORD
                  nParam  Posint;
                  fromParam  char;
                  toParam  char;
                  x nParam  ar
                END;
      stake= RECORD
               to  0..maxstack
               item  array[1.. maxstack] of Dataarea
             END;
VAR s  stack;
    currarea  Dataarea;
    un  Boolean;
    temP  char;
BEGIN
    (*  inicijalizaci a  *)
    s.toP =0;
    currare  nParam =n;
    currarea.fro  ram =fromPe9;
    currarea.toParam=t Pe9.
    currarea auxPa  m=auxPe9;
    (* Pocetak si  lirane rutine *
    IF curr re  par  ei
     THEN
      Wi H  rarea"
       BEGIN
        writeln  Prenasi  isk  f  sa  staPa .fromParam,
                  ne staP  toP  am)
         * simulacija Povratk  )
        P  =  t s cur area
        IF
          THEN GOTO 1 (* Povratak  Glavni Program *)
          E SE GOTO 2  * s ok na ta ku izz r k. Poziva *)
       END  * then with *)
    * Pr  :  e ur  =  Pozi  *
    Push(s c rrarea
    Ok ra  a nParam =s.itemEs.toP].nParam-1
    currarea fro Param =s.ita Es toP].f  P  am,
    currarea toParam =s itemE toP].a xPara
    curre ea a xParam =s itemEs.to ].  oParam;
    GOTO 1
  * eve se  racamo in pr jce rekurzivnog Poziva *)
    itel ( P  e i Di k  c  a  a.n ra  a  t Pa'
           c  rarea.fromPara  ' a staP  cu  area toParam);
     Drugi r  r  v i Poziv *
        ea.nPara  =cu  a  a. Param-1;
    nnP =c rarea  =Param
        a fro P ram =c rrarea  uxParam
     rP  a au Param =temP
    GOTO 1e
END (* simtowers *);
```

slika 3

(4) skok na početak simulirane rutine
Nakon što se simulirana rutina izvrši, spremne je za povratak Akcije idu sledećim redom
(5) Sečuva se adresa povratka, i, iz tekuće oblasti podataka, a2
(6) Uzima se gornji element se steka i dodeljuje se tekućoj oblasti podataka, a1
(7) Skok ne 1

Uočimo da je I, u stvari iebele od end Dekle sledeći korak je još jedno uzimenje elementa se steka i još jedan povratak Nikade ponovo ne koristimo informeciju iz tekuće oblasti podataka e1 jer je ona odmeh uništene uzimenjem novog elemente sa steke Dekle nemerno razloga de pri simulaciji poziva tu obiset podataka uopšte stavljemo na stek

Poziv dakle, možemo da simuliramo jednostevno kso
(1) Promene parametra u tekućoj oblasti podataka na odgovarejuće vrednosti

(2) Skok ne početak simulirane rutine

Pri povratku iz simulirana rutine možemo se vratiti direktno u rutinu koja poziva našu proceduru jer neme koristi od toge de to radimo u dva uzastopna poluskoka

Ostela nam je još samo jedne moguća adresa povratka pa

```
PROCEDURE sintowers(n  Posint; fromPee toPe9.auxPe9: char
CONST maxstack=55;
TYPE  Dataarea= RECORD
                  nParam  PosinP
                  fromParam  char
                  toParam  Char
                  auxParam: char
                END;
      stack= RECORD
               toP  0..maxstack
               item  ARRAY[1..maxstack] of Dataarea
             END;
VAR s  stack;
    currarea  Dataarea;
    un  Boolean;
    temP  char;
BEGIN
    (*  inicijalizacija  *)
    s.toP =0;
    currare .nParam =n;
    currarea.fromParam:=fromPe9;
    currarea t  a am =to Pe
    currarea .auxParam =auxPe9;
    un =fal e
    REPEAT
     WHILE currarea nParam^)t DO
      WITH cur ar a DO
       BEGIN
        PushK  ;  rrarea)
        nParam:=s.itemEs.toP].nParam-1.
        fromParam =s.itemEa toP].fromParam,
        toParam =s.itemEs toPJ a  Param;
        auxParam =s itemEs.toPJ.toParam;
       END (* Do begin *)
    writel  Pre e i Disk I sa staPa  c rra ea fro
                 r  aP  currarea.toParam
    PoPanete t  rrarea un)
    IF NOT un
     THEN
      WITH cu  area DO
        BEGIN
          writeln( Prenasi Disk  oP ram  sa staP  f
                    na staP  toParam)
          nParam =nParam-1
          temP =fromParam;
          fromPaam =auxParam
          a  Pa am =temP
        END    with *)
    UNTIL un
END (* simtowers *);
```

slika 4

je ne moremo spremati i neknadno uzimati se steke se ostelim podecime iz oblasti podataka Kad god uspešno izvršimo uzimenje elementa se steke previmo skok ne adresu neredbe neposredno iza prvog poziva Ako pokušaj uzimenje se steka javi underflow skočićemo u pozivni program Keko se nova vrednost promenjijvih u tekućoj oblasti podataka dobije iz stare oblasti podeteka moraćemo da deklarišemo dodetnu promenjivu *temp* koje će privremeno čuvati vrednost stare oblasti podeteka

Poprevljene nerekurzivne verzije hanojskih kule date je na slici 3

Ni ovo nije savršene verzije Ako se melo pozabevimo strukturom programe leko čemo ga reorganizoveti Jednostavniji format hanojskih kule dat je ne slici 4

Ovo je sasvim pristojan oblik iterativne procedura Znači , kupili smo rekurzivno rešenje probleme „meheničkim" ga prevell u nerekurzivni oblik , ne kreju to rešenje „očistili od višake Navedenim postupkom uradili emo ono što je u početku zvučalo preteško — imamo nerekurzivno rešenje zadetake eko emo bejzikoljupci leko čemo ga prepevati na bezik Ček i eko i deije ostanemo u paskalu, dobili smo rešenje koje se brže izvrševa od rekurzivnog enalogona

Zoran Obradović

Programiranje na mašincu

Linkovi na „spektrumu 128"

sinklerova veza

Meda to ne ići na njega. Sinkler je uveo u ROM „Spektruma 128" dosta linkova za proširenja (iako se autoru čini da je on to uradio zato što nije mogao drugačije a ne da nema oteksa život). Naime veza između dva stranica u ROM-u odvija se preko izvjesnih linkova u RAM-u. Prije nego što nastavite dalje čitati tekst preporučujem vam da pročitate tekst u prošlom broju „Računara" jer vam u suprotnom ono o čemu se ovde govori neće biti najjasnije.

Prijavljujem grešku!

Kako definisati nove naredbe na „spektrumu 128". Prvo što je potrebno to je da se upozna mehanizam prijave greške. U ROM-u 1 greška se prijavljuje kao što svi znaju, pomoću

```
RST 8
DEFB kod greške
```

dok se u ROM-u 0 greške prijavljuju pomoću

```
CALL 1452
DEFB kod greške
```

Dakle, u ROM-u 0 prijave greške nije dobila svoj restart, ali to ništa ne mijenja stvar. Princip prijave greške je isti i svodi se na

```
LD SP (23613)
RET
```

All dok je kod 48-ice varijabla ERR_SP (23613) ukazivala na adresu 1303 heksadekadno ili 4867 dekadno, kod 128-ice ukazuje na adresu 23325! To je u RAM-u zar ne? Baš lijepo! Upravo ta mogućnost nam dopušta da radimo šta hoćemo. Tamo sa nalazi sljedeći mašinac

```
DI
LD A, (23388)
AND 0111%1111
LD (23388), A
LD SC, 32765
OUT (C), A
EI
JP 195
```

Ko je čitao dosadašnje tekstove o „spektrumu 128" trebalo bi da shvati šta radi ova rutine. One prvo uključuje ROM 0 i vrši skok na adresu 195 (a ovo je već ROM). Tamo se nalazi ova vrlo kratka rutina

```
LD HL, (23435)
JP (HL)
```

Vi ćete pomisliti: sve je jasno, samo treba na adresu 23435 smjestiti neku vrijednost i kontrola prijave greške će biti usmjerena na našu rutinu. All ne, nije tako! Ova sistemska varijabla ima i rutinu osobinu: da se mijenja u toku rada. Drugim riječima za vrijeme izvršavanja programa (run-time) one ima vrijednost 601 dok je za vrijeme sintaksne provjere tamo neka druga vrijednost najčešće 693, all ne uvijek. Na taj način kompjuter razlikuje sintaksne od run-time grešaka. Ova sistemska varijabla može vrlo dobro poslužiti za run-time rutine tipa ON BREAK GO TO, ON ERROR GO TO itd. All nama one ne odgovara. Zato ćemo uraditi nešto drugo: promijeniti instrukciju JP 195 u JP naša rutina. Jednostavno, mora se priznati!

Sa teorije na praksu

Na jednostavnom primjeru ćemo pokazati kako to izgleda u praksi. Prije toga, riječ-dvije o ROM-u 0. Restarti RST 16, RST 24 i RST 32 su identični onima u ROM-u 1, preciznije, oni pozivaju istoimene rutine u ROM-u 1. Restarti RST 8 i RST 48 ne postoje a šta radi RST 0 i RST 56 to se zna (pogotovo RST 0). Zaboravne podsjećamo da se RST 40 koristi za pozivanje rutine u ROM-u 1 i to u obliku

```
RST 40
DEFW adresa rutine
```

Nove sistemske promenljive

U Računarima 19 spomenuli smo neke nove sistemske varijable. Ovdje navodimo kompletan spisak sistemskih varijabli sa potrebnim objašnjenjima.

DUŽINA	ADRESA	IME	OPIS
N2	23384	NL–SAV	Služi za skladištenje HL registra u slučajevima kada primjena steka nije poželjna
X2	23386	RET–ADD	Adresa povratka u RM 0 (vidi tekst)
X1	23388	MEM–ST	Sadrži stanje memorije. O ovoj varijabli bilo je dosta riječi u prošlom broju
N2	23389	E–HANNLE	Ova dvije lokacije sadrže rad m 397 (RST 8) kod greške Kad se kod RM 0 prijavljuje grešku z RM-a 1. Za nas je ova varijable potpuno neupotrebljiva.
	23391	BAUD	Brzinska konstanta za RS 232 operacije. Mijenja je naredba FORMAT
2	23393	SER–FLG	Statusna var jabla za RS INPUT operacije
N2	23395	RS–COUNT	Brojač prazn h karaktera prilikom ispravnje kontrolnih kodova 8,22 i 23 na RS 232 izlaz.
1	23397	RS–CTRL	Broj parametara kontrolnih kodova
X1	23398	T–FLAG	Razn ind kator za operacije sa RAM diskom. Ova a može imati primjenu kod pozivanje RAM disk rutina iz maš nca
N10	23399	NAME	Ime zapisa koj se poziva iz RAM diska
N15	23409	HEADSRS	Radni prostor za operacije sa RAM diskom. Također se koristi i prilikom naredbi RENUMBER, FORMAT. Ako se pomenute naredbe ne koriste, ovdje se mogu smjestiti razni podaci
N2	23425	ŠP–SAV	Čuva staru vijednost SP registra prije njegove promjene
X2	23427	LAST–FILE	Ukazuja na zaglavlje posledn je datoteke u RAM disku (vidi: Računara br. 19)
X3	23429	FREE–DISC	Broj bajtova slobodne memorije u RAM disku (PEEK 23429+PEEK 23431*256+256*PEEK 23431 [5536]. NAPOMENA: Zbog bага u ROM u, ova sistemska varijabla totalno poludi ukoliko zapišete datoteku dužu od 65527 bajtova (a kome treba tolika datoteka???)
3	23432	PAD–SCAN	Koristi se za očitavanje KEYPAD-a
X2	23435	ERP–PTR	Pointer rutine za prijavu greške
N5	23437	RES–CALC	Rezu tat posledn je operacije u kalkulatoru.
N2	23442	REN–PTR	Ukazuje na liniju čija je prenumeracija u toku.
6	23444	REN–LINE	Početna linija za RENUMBER
2	23446	REN–STEP	STEP za RENUMBER
N15	23448	GRAPH–CHR	Ovdje se nalaze definicije 2 grafička karaktera koji služe za ispisivanje menija
N80	23464	STACK–BUFF	Služi za smještanje mašinskog steka za vrijeme operacije koje preklapaju gornjih 16K memorije (editor LOAD' itd.)
	23552–23733		Stare sistemske prom ve

Oznaka N iza broj dužine var abla znači da pok ranje te promjenjive neće izazvati nikakav efekat a oznaka X upozorava da sistem može da krahira ukoliko se u varijablu smjeste nedozvoljene vrijednosti.

Imena sistemskih varijabli su plod pjesničke mašte autora te se nemojte čuditi ukoliko u nekom stran m časopisu naidete na nešto sasvim drugo. Međutim, bar koliko je autoru poznato kompletan popis sistemskih varijabli nije dosada nigdje objavljen.

Postoje i neke sistemske promjenjive koje se nalaze izmješane sa radnim prostorom editora u RAM-u 7. Međutim više se pole po nutih varijabli ima namjenu u stilu: trebam negdje da smjestim pod tka tek-m n ga ih ovdje nećemo opisivati. Detalji služe za opisivanje raznih stanje pri radu editora, ali ni od njih nema neke velike koristi. Autor smatra da su najkrisnije od njih 60433 i 60431, koje su opisane u prošlom broju.

Svaki haker koji je bar malo upućen u tajne mašinskog programiranja zna da je definisanje novih naredbi ne „spektrumu 48" veoma mukotrpan ali ipak mogućan posao. Naime, Sinkler nije smatrao da će vlasnicima 48-ice ikada „zatrebati" tako nešto. Na svu sreću, I to se može nekako ostvariti (IM 2, ERR-SP), ali pravi problem nastaje kada treba izmijeniti neku već postojeću naredbu (npr. LOAD). U tom slučaju je jedini ilijek bilo pisanje potpuno novog bejzik interpretera. „Spektrum 128" predstavlja, srećom, sasvim drugu priču.

Pomoću rutina se prijavu greške u ROM-u 0 (adresa 1452) mogu sa prijavijivati i nove greške Dna imaju kodove 28–43 i idu slijedećim redom a MERGE error, b Wrong type file, c CODE error d Too many brackets, e File already exist, f Invalid name, g File does not exist, h File does not exist i Invalid device, j Invalid baud rate, k Invalid note name, l Number too big m Nota out of range n Out of range, o Too many tied notes Autoru je najsano kako se dsalio da postoja dvije iste greška (kodovi g i h) I da postoji „nova-stara" greška (kod 1).

No, catavimo Sinklerova biserа po atrani i vratimo se našem zadatku Recimo da želimo definisati naredbu JOIN koju „spektrum 128" na poznaje, a vrlo je korisna Naime, ona spaja dvije bejzik linija u jednu, što je normalnim putem teško catvariti

Počnimo Ukotiko želimo da se naša naredba (racimo JOiN) može koristiti i u komandnom i u programskom modu, ona mora početi nekim znakom koji nije slovo U suprotnom, pošto u komandnom modu PRINT naredba nija potrebna postoji opasnost da naša naredba bude shvaćena kao varijabla (probajte na prazno otkucati JDIN 10 I dobićete „2 Variabla not found) Zato ćemo odlučiti da se naša naredba zova racimol JOIN III još bolje samo !J (lakše je za kucanje kao ima manja stova) Rutinu ćemo smjestiti na adresu 65000 I ona izgleda ovako

```
DRG 65000       početak na 65000
LD A, (23610)   uzima kod greška
CP 11           da li je Nonsenca in BASIC?
JP NZ, 195      skače u RDM ako nije
RST 24          uzime karakter
DEC HL          na kom je
LD (23645), HL  nastupila
LD A (HL)       greška
CP  !           da ti je to „!"?
JP NZ, 195      ako nije  skače u RDM
RST 32          uzima slijedeći karakter
AND 223         pretvara malo slovo u veliko
CP „J"          da ti je to „J"?
JP NZ, 195      skok u ROM ako nije
```

Dvim je završen prvi dio našeg posla Nakon što je detektovana naša naredba, potrebno je na atak postaviti adresu za prijavu greške (23325) I povratnu adresu u bejzik interpratar (to je rutina ŠTMT–RET) U atarom bajzik intarprateru to ja bila adresa 7030 a u novom je 8117 Dakle, slijedi

```
LO HL  23325    na atek
PUSH HL         adresa za prijavu greške
LD HL  8117     na atek povratna adresa
PUSH HL         u bejzik interpreter (novi)
```

Dalje naša naredba ima jedan numerički paramatar (to će biti broj linije koja sa spaja sa slijedećom) Njega ćemo pročitati pomoću rutina EXPT–1NUM (adresa 7296) u ROM-u 1 koja „akida jedan numerički paramatar (pogledajte SPECTRUM ROM DiSASEMBLY od 48-ica ako ga imate) Dakle

```
RST 32          uzima slijedeći znak
RST 40          I poziva rutinu
DEFW 7296       EXPT–1NUM iz RDM-a 1
```

Pošto nama više paramatara treba provjeriti kraj Stijedi

```
RST 24          uzima karakter
CP 13           da li je to ENTER?
JR Z  DALJE     ako jeste, skok na labelu DALJE
CP „,"          da li je to „,"?
JR Z  DALJE     u redu je ako jeste
CALL 1452       u suprotnom
DEFB 11         slijedi greška
```

Ako je sve u redu, treba signatizirati da nema greške, i ukoitko je bila u toku sintaksna provjera, posao je gotov Dakle u nastavku stijedi

Kako prepraviti mons

Jedna od zanimljivih osobina spektruma 128' je i ta da se lati ROM koristi i iz 48 I iz 128 moda (jedino je u 48 modu pola ROM-a isključeno) I zato veliki broj programa radi u oba moda Čak sa i čuvani GENS za 48K „spektrum može koristiti i iz moda 128

Međutim, sa MDNS-om to nije slučaj Ovaj program redi sasvim lijepo u 48 bejzik modu „ati u 128 modu ima probleme sa povratkom u bejzik No, dovoljno je par „pokova pa da sa I to riješi Ukucajte stijedeći program

```
10 LOAD „MONS3M' CODE 30000,5800
20 POKE 30089,3  PDKE 30070,19
30 SAVE „MDNS 128  CDDE 30000,5800
```

Nakon ovoga MDNS možete bez problama koristiti i iz bajzika 128 ukoliko je to potrebno (npr da iztatate naki mašinac na RS 232 izlaz)

```
DALJE LD
(tY+0), 255     signal „nama greška
BIT 7, (IY+1)   da li je bila u toku sintaksna provjera?
RET Z           kraj ako jeste
```

Ukoliko nija bila u fazi provjera sintakse nago izvršenje (run-time) treba samo izvršiti nareddbu i to je ava Pošto bi JOIN rutinu trabalo da zna napisati svaki haker koji čita ovaj tekst ovdje je dajemo bez objašnjanja

```
JDIN RST 40
     DEFW 7833      ovo je rutina FtND–iNT2
     LA A  B
     CP 64
     RET NC
     LD H, B
     LD L, C
     RST 40
     DEFW 6510      rutina LINE–ADDR
     RET NZ
     RST 40         rutina NEXT–ONE
     DEFW 6584
     LD A  (DE)
     CP 64
     RET NC
     EX DE  HL
     DEC HL
     LD (HL),
     iNC HL
     PUSH HL
     INC HL
     tNC HL
     LD C, (HL)
     INC HL
     LD B  (HL)
     EX DE  HL
     INC HL
     iNC HL
     PUSH HL
     PUSH BC
     LD C  (HL)
     tNC HL
     LD B  (HL)
     PDP HL
     ADD HL, BC
     EX DE  HL
     POP HL
     LD (HL)  E
     INC HL
     LD (HL)  D
     POP HL
     LD BC, 4
     RST 40
     DEFW 6632      rutina RECLAiM–2
     RET
```

Dakle, proces definisanja novih naredbi na spaktrumu 128 donakle je sličan procesu definisanja naredbi na 48-ici s tom razlikom da nama mučnih priprama oko tM 2 ERR–SP i sličnih „gluposti Jedino se ne smija smesti sa uma da je cijelo vrijeme dok traja proces definisanja novih naredbi uključan ROM 0

Nova naredbe je potpuno ravnopravna sa catalim naredbama i može sa koristiti bilo u programskom bilo u komandnom modu. Npr da spojite liniju 10 sa slijedećom linijom samo otkucajte

IJ 10

Ukoliko linija 10, ne postoji ništa se neće desiti

Umalo da ne zaboravimo. Da bi interpretar uopšte mogao da prepozna novu naredbu, treba „prepokrati" JP 195 u JP 65000, tj izvršiti

```
POKE 23341, 65000-256*INT(65000/256)
POKE 23342, INT(65000/256)
```

Tajna restarta 40

Činjenica da instrukcija RST 40 „petlja" se ROM-om daje nam mogućnost da utičemo ne njano izvršenje, a zamim tim i na izvršenje mnogih već postojećih naredbi i rutina. Ovaj restart izgleda ovako

```
EX (SP), HL     na stek HL, u HL adresa iza RST 40
PUSH AF         sklanja AF
LD A, (HL)      uzima bajt iza RST 40
INC HL          uvećeva HL za 2
INC HL          i smješta rezultat u
LD (23386), HL  sistemsku varijablu RET-ADD
DEC HL          umanjuja HL za 1
LD H (HL)       u HL registru se sada nalazi
LD L A          adresa koja se poziva preko RST 40
POP AF          vraće AF
LD (23384), HL  privremano sklanja HL
LD HL, 23316    ne atak ida 23316, a u HL
EX (SP) HL      se vraća stara vijednost
PUSH HL         atavija HL na stek
LD HL, (23384)  vraća HL (u njemu je adresa rutina)
EX (SP) HL      izvršne adresa ida na atak, a u HL stare vrijabla
JP 23296        akok u RAM
```

Ova rutine, varovatno, izgleda dosta komplikovano zbog nezgrapnih operacija se atekom ali, u auštini, ona ne radi ništa strašno One samo na atek dovodi adresu 23316 i adresu rutine koju pozivamo preko RST 40, a adresu povratka u ROM 0 smješta u RET-ADD (23386) Pri tom vodi računa da na „pokvari" niijedan registar i to ja sva

Nakon toga slijedi skok u RAM na adresu 23296 Rutina koje se nalazi tamo je krajnja jednostavna i njena uloga ja da preklopi ROM Rutina izgleda ovako

```
PUSH AF         sklanja AF
PUSH BC         sklanja BC
LD BC, 32765    ovo je kontrolni port za stanja memorije
LD A, (23388)   čita stanje memorije
XOR %00010000   „izvrće" bit za ROM
DI              isključuje interapt
LD (23388), A   pamti novo stanje memorije
OUT (C), A      i preklapa ROM
EI              uključuja interapt
POP BC          vraća BC
POP AF          vreće AF
RET             kraj
```

Prilkom nailsaka na RET vrši se akok na rutinu koje ja pozvana sa RST 40 (ne zaboravite šta ja na ateku) Nakon ailnih paripetija tražena rutina je najzad pozvana A šta se dašava posla?

Kada se pozvana rutina završi RET instrukcija na njenom kraju izaziva akok u RAM na adresu 23316 (opet zaeluga ateka) Mašinska rutine koje se tamo nalazi vrlo ja kratka

```
CALL 23296      preklopi ROM
PUSH HL         sačuvaj HL
LD HL, (23386)  uzmi povratnu adresu
EX (SP), HL     stavi povretnu adrsau ne atek i vrati HL
RET             akoči nazad na mjesto odakis je rutine
                blis pozvana
```

Ovim je tajna restarta 40 objašnjana.

A ŠTA SAD???

Svi koji au ahvatiii kako radi pomenuti restart već imaju predstavu kako se on može iakoristiti Mogučnosti au zalata ogromne, a ovdja ćemo dati jedan primjer

Pretpostavimo da želimo da nepravimo novu POKE rutinu koje će prihvatati i adresa veće od 65535 i recimo da proglasimo da su adrese od 65536–147455 adresa RAM diska To nija prevalik problem Pomoću ta rutine možemo zamisliti da imamo 16-bitni procesor (opa!) i zaboraviti da nam je memorija organizovane „po blokovima" No, predimo na atvar

54/programiranje na mašincu

Novi bajzik interpreter se nalazi u ROM-u 0 (za radoznalca to je adresa 6063) a POKE rutine u ROM-u 1 na adresi 1E80 HEX (vidi ROM DiSASEMBLY) Logično je da bajzik interpreter NEGDJE (gdja, to uopšte nije važno) poziva POKE rutinu u ROM-u 1 pomoću

```
RST 40
DEFW 1E80 HEX
```

Priča počinje ovdje Na adresu 23296 ćemo amjestiti JP 40000 (može i neka druga adresa ali zbog prirode rutine ta adresa bi trebala da buda iapod 49152) Na toj adresi prvo treba uraditi sva ono što je redila rutine na 23296 dakle

```
ORG 40000       DI
PUSH AF         LD (23388), A
PUSH BC         OUT (C), A
LD BC, 32765    Ei
LD A, (23388)   POP BC
XOR 16          POP AF
```

Dalje, potrebno je „pogledati" koja je rutine pozvana. No, moramo dobro paziti da ne „umrljamo" ni jedan register (pa ćek ni F koji ima ružnu osobinu da se kvari i kad treba i kad ne treba). Ukoliko nija pozvane POKE rutina, ne treba da se desi ništa specijalno. Madutim, ukoliko je pozvane POKE rutina, treba da uslijedi skok ne našu rutinu To ava izgleda ovako

```
                EX (SP), HL     sklanja HL i uzmi u HL
                                adresu se steka
                PUSH AF         sklanja AF
                LD A, H         uzima viši bajt
                CP #1E          da li je to #1E?
                JR NZ, NO-POKE  ako nije, skok ne NO-POKE
                LD A, L         uzima niži bajt
                CP #80          da li ja to #80?
                JR Z, POKE      ako jeste, skok ne POKE
NO-POKE         POP AF          vraća AF
                EX (SP), HL     vraća HL i adresu skoka
                RET             skače na pozvanu rutinu
POKE            POP AF          vraća AF
                POP HL          uzima HL i, ujedno, poništava
                                adresu skoka
```

Dalje treba da slijedi nova POKE rutina. Pošto nam ovdje nije bio cilj da objašnjavamo kako nepreviti novu POKE rutinu, nego kako izmijaniti postojeću naredbu, ovu rutinu vam ostavljamo za vježbu

Kao što vidite, izmjene rutina u ROM-u je vrlo jednostavna. Krajnji efekat zavisi samo od mašte hakera, Dakle, GENS u ruke (ako imate MAGUS još bolje) i na posao

Umjesto epiloga

No, mora se reći da postoji i jedan problem ROM 0 na poziva baš svaku rutinu iz ROM-a 1 U prošlom broju je apomenuto ne koje se naredba može u potpunosti djeiovati na ovaj način Što se ostalih naredbi tiče, njihove izvršne rutine se nalaze u ROM-u 0 zajedno se novim bajzik Intepreterom To je urađeno zbog toga što au neka rutine pretpiia izvjesna izmjene Ipak i te rutine pozivaju neka podrutine u ROM-u 1, tako da ja i na njih moguće donekia utiosti Npr rutine LOAD lako je amještene u ROM 0 (zbog opcija „!") one i dalje poziva rutinu LD-BYTES (1366) u ROM-u 1: Jadino se ne može utiosti ne rutine koje au u potpunosti u ROM-u 0 (ne žalost tamo spadaju PLAY, COPY i još neke zanimijiva rutina)

Doduše postoji način da se i to aradi Na primjer, rutina PLAY ipak poziva nešto iz ROM-a 1 (to je rutine za skidanje alfanumaričkih paramatare), te nam i to otvara mogučnost petljanja" No, o toma drugom prilikom Uostalom, možete i sami pregledati rutine bajzik interpretera koje au prilično „čitljiva U avakom slučaju alabo dokumantovan računar kao što je „spektrum 128", daje nam idealnu prilku za „čeprkanja"

I još nešto za kraj ROM 1 ja lako „ispikirati" iz bajzika, ali šta je se ROM-om 0? Ako se još niate ajetili kako da „ispikirata ROM-0 najboije to možete uraditi ovako

```
SAVE !„rom 0  CODE 0,16384
LOAD !„rom 0" CODE 32768
ERASE !„rom 0"
```

Ovim je ROM 0 prebačen u RAM na adresu 32768 i možete ga gledati do mila voija Šta čekate!?

Željko Jurkić

algoritam „brzi gonzales"

Po ko zna koji put se potvrdila činjenica da imamo vrsne pronalazače i stručnjake na svim poljima tehničke djelatnosti, ali i da još uvijek imamo „tvrd" sluh kada je u pitanju svestrana podrška pronalazača. Kako drugačije komentirati riječi dipl. ing. Željka Margitića koji kaže: „Radim na sistemskom softveru preko 14 godina. Međutim, više od 10 godina morao sam raditi u vlastitoj režiji, jer mi moja radna okolina nije pružala podršku u radu niti je za njega imala sluha."

— Još daleke 1975. sam imao gotov sort koji je radio dvostruko brže od Singlarovog Narevno sve je to komisijski potvrđano i dokazano Godinama sam krpao tuđa mozgarija te sam 1983 odlučio razviti visoki sistem isposlovim sa da je ono što sam napravio na Singlarovom SISTEMU 10 daleko bolje od postojećih rješenja Najbolji rezultati se postižu kod random pristupa na datotekama i to zahvaljujući novoj DIRIN metodi koju nama još ni jedan današnji računar jer sam napravio algoritam za pri stup disk adresama duljina svega 5 karaktera Ova metoda je brža od bilo koje random metoda do seda s glavne datoteke ostaju uređena se maksimalnom uređenosti prostore na disku Na osnovi DIRIN metode učitaju se kompletni DIRIN indeksi u memoriju koja se sastoji od 4 bajta disk adresa za MC 68000 — 16 bitni mikroprocesor (izbor izvršio doc dr Andrija Marličić sa ETF-a) + 1 bajt za status zauzetosti fleksibilna baze podataka

Od nemila do nedraga Dipl ing Željko Margitić

Fleksibilna baze podataka

Što podrazumijevate pod FLEKSIBILNOM bazom podataka?

— Da bismo objasnili prirodu fleksibilne baze podataka uzet ćemo njezin način funkcioniranja na obradi seida kontija kupeca i dobavljača Npr poduzeća s oko 2000 ljudi imaju register kupaca s oko 30 000 slogova Ako uzmemo algoritam gustina 4 na jednu stranicu da prosječno otpadati tri sloga datoteke registar Ako uzmemo dirin indekse u memoriju oni će zauzeti oko 60 K DATA BASE MATE mašina će moći vrtiti u memoriji oko 100 programa Koncepcija mašine je multi task, multi proces multi user Nema ograničenja da svih 100 programa ima ju dostup do registara datoteks Pristup registar slogovima ide od svih programe preko učitanih dirin indeksa u memoriji Program koji traži neku šifru „zalokuje cijelu stranicu registra (4 kupca) i drugi program nema dostup do te stranice sva dođe dok prvi program ne obavi posao na toj stranici (transakcije)

Za to vrijema drugi program radi isti posao na nekoj drugoj stranici s n-ti program na n-toj stranici Algoritam kinematika diska (razredio ing Margitić o p) vodi računa o toma da se posao što prije obavi na cijelom računaru

Što se zapravo, događa u računaru?

— Kada jedan program zatraži pristup do nekog kupca pomoću recepi algoritma koji je se svaku datoteku tiekslbilno napravljen uz vođenje računa o analizi karakteristika šifre povezano s dva algoritma u funkciji dva nivoa sebljenja prostore disk adrese početka stranice se nađu u memoriji procesora Nakon toga slijedi direktan skok na traženu stranicu se slogovima bez prethodnog pretraživanja na disku (to do sada niko nije postigao vrijema ICL-a na SISTEM 10 računaru je 2 minuta i 26 sekundi s dirin algoritma 29 sekundi, što je oko 5 puta brze o p) ‚MATE' računar će biti prva preva deste base mašina u svijetu se stanovišta da data base obrada budu ubrzana na svim nivoima upotrebe — batch obrada upiti teleskomunikacije itd Već kod dizajniranja arhitekture računare treba o svemu ovome voditi računa, kao i o ‚Z kompajleru' koji prati cijelu arhitekturu računare, e koji je daleko moćniji od C kompajlere Zahvaljujući takvoj obradi ‚MATE softvera omogućava se blok random pristup podscima Algoritam kinematike diska omogućava kupljenja blok slogova podsteka u jednom potezu ručice uz optimalizaciju rotacije diska što daje još veće ubrzanje dirin metodi

Računar otvorenog tipa

Da, ali što će se desiti ako npr nestane struje?

— Ukoliko nestane struje ‚MATE' računar omogućava restauraciju rade svih 100 programa u kompjutoru i to točno na onim pozicijama gdje je nastao prekid i ako je memorija nakon prekida struje izgubila svaki sadržaj i to bez obzire na vrijeme trajanja prekida To se do sad nije nitko dosjetio riješiti, a može se omogućiti sottverskom evidencijom poslova i tekova na takav način da se uopće na osjeti usporenja rada mašina pa nisu potrebna nikakva prethodna opasavanja datoteka zbog naknih osjetljivih ažuriranja

Na koji način će doći do potpune realizacije projekta?

— Prije izvjesnog vremena je došlo do povezivanja GOUR-a Tesre (RO Tehnicer) Elektronike' (RO ‚Pel) ‚Zavoda za ekonomiku iz Varaždina i mene ta je senavan „RIIC (razvojno istraživačko informatički centar) također se sjedištem u Varaždinu U toku su dogovori Pela i ‚Tere oko razvoja elektroničke logike po principu ‚MATE' operativnog sistema usavršavajući ‚Tera 6 računar koji je bazirn na MC 68000 sa ST 506 disk kontrolirima koji u svom momentu omogućuju priključke diskova od 250 MB ‚Tera 8 ima mogućnost da CP se MC 68000 ima 8 MB memorije Za CP se 32 bitnim mikroprocesorom MC 68020 bit će pradvidena memorija od 16 megabajta Svake disk jedinice će imati svoj visetiti mikroprocesor i to prvi visetiti u našoj zemlji (sve su do sada bile iinosnce o p) a leto tako se planira da svi periferali budu s visetitim mikroprocesorima Teži se ka toma da se projektuje razvoj otvorenog tipa računare čima bi se omogućio saobraćaj svih računskih centara i drugih poduzeća u zemlji u planiranoj JUPAK i ŽEPAK mreži Saobraćaj bi se ostvarivao putem telefonskih linija i to zahvaljujući mogućnosti multi pristupa na sve datoteka Koncept MATE razvoja ide za povezivanjem svih svojih korisnika i za međusobno koordiniranja i preuzimanja programa, čima bi se povećala produktivnost izrada i obrade aplikacija u cijeloj zemlji te će se u potpunosti moći pratiti razvoj transpjutora, ojačati obrambene sposobnosti JNA i ubrzati tokovi razvoja mnogih privrednih grana.

Deset puta jači, deset puta jeftiniji

Koja je cifra aktualna kada su u pitanju ulaganja?

— Ako se posmatre dobit od jednog takvog sistemskog softvera onda su troškovi ulaganja praktično zanemarivi Potrebno je oko '5 milijardi starih dinara U tu cenu su uračunsta tri razvojna sistema bazirena na ‚Teri 6 poslovni prostor plaća za 10 razvojnih inženjera otkup linosnenog prava (od ing Margitića o p) Jedna takva gotova mašina bi na tržištu koštala oko 10 milijardi starih dinara s bazom podataka a već nulta serija bi sapetila troškova i donijala zaradu za proširenu reprodukciju Da kažem i to da je općina Varaždin spremna nevažno pomoći čitav projekt jer u toma vidi svoju perspektivu s je pozivam sva one koji hoće da sudjeluju u razvoju „RIIC -a da nam se priključe jer još nije kasno Nedavno je RO Rads Končar se procesni računar ASEA pisto 3 mil $ s MATE računar će biti 10 puta jeći i efikasniji od ASEA i svega 10 puta jeftiniji Dodajmo tome i da smo prije par dana uspostavili kontakt se sva vrsna makedonska pronalazača Gruevskim i Nured nom te je velika vjerovatnost da algoritam za rješavanje desetog Hilbertovog problema po prvi put na svjetsko tržište bude plasiran putem „MATE računara.

Na kraju nagiselmo da se kompjutori samo srodstva za rad. s se dinamičnu informatizaciju modernog društva dominantan je aspekt razvoja softvera Godišnja proizvodnja softvera u svetu raste za oko 17% Vrijednost podinjaj plasmana softvera na svjetsko tržište se kreće između 7 i 12 mird $ a posiljednjih godina se javlja novo tržište mikrokompjuterskog softvera koje se za 2000 pod procjenjuje na 20 do 25 mird $ U ovu proizvodnju su se već uključile mnoge zemlje u razvoju budući da su ulaganja neznatna a osnovni kapital predstavlja ZNANJE Po svemu sudeći, svijat će uskoro dobiti veliku konkurenciju iz drugog plana, iz Jugoslavije

Dragan Vranješević

iz šupljeg u prazno

Sortiranje (2)

Znate li onaj vic sa bradom o matematičkim nastrojenom programeru koji spreme čaj? Programer, da vas podsetimo, zna kako da spremi čaj: sipa vodu u kotlić, stavi kotlić na šporet, sačeka da voda provri i tako dalje. Šta će programer da uradi ako mu neko de kotlić u koji je već sipane voda? Prosuće vodu i time rešiti problem na prethodni sličan koji ume da reši.

Možemo li da sortiramo podatke sa spoljne memorije tako što ćemo proeuti vodu tj. učitati ih u osnovni RAM i tamo ih sortirati, koristeći neki od metoda koje smo upoznali pre mesec dana? Ako podataka ima malo, ovakvo vam rešenje toplo preporučujemo. Podatake, međutim, obično ima previše, pošto je kapacitet spoljne memorije obično nekoliko desetina (stotina hiljada) puta veći od kapacitete RAM-a, eve se podataka ne mogu učiteti i teko sortirati. Ostaje dakle da upoznemo metode koje omogućavaju sortiranje velikih količine podataka koji su upisani ne treku, flopi ili masivni disk.

Sekvencijalne memorije

Magnetne treka je, istorijski posmatrano, prvi medij masovne memorije, nekada je bilo sasvim normalno da (veliki) kompjuter bude opremljen jednim diskom kapaciteta pola megabajta i **trima** jedinicama magnetne treke: možde ćete zbog toga u mnogim neučno-fentastičnim filmovima primetiti bezbroj treke koje se okreću na razne etrana.

Jedinice magnetne trake nisu nimalo jeftine: uobičajeno je da koštaju više od standardnih jedinica hard diskova! Zbog čega ovolike cene? Jedinica magnetne treke ima **izuzetno** preciznu mehaniku (i pneumatiku) koje omogućave de se treke iz brzog premotevenje praktično trenutno zaustavi i teko omogući direktan pristup podacima koji nisu upisani ne njen početa. Trake se, osim toga premotave ne obe strane što znači de u izveenoj meri može da se smetra medijem se slobodnim pristupom.

Pad cene dlekova je učinio de ovakve skupe i super precizne treke izgube ne značaju, disk je deleko brži i zgodniji medij na kome je direktni pristup mnogo direktniji. Zašto su onda treke i dalje popularne? Zato što cela kede kupite (ekupu) jedinicu treke u nju umeteti magnetne treke koje su izuzetno jeftina i koje se leko skladište, zauzimaju malo prostore ne treba im posebna klimatizacije prešine obično nije beš smišljena. Podatke, dakle, obrađujemo držeći ih na disku, a zatim detoteke koje treba skladištiti sekvencijalno prapisujemo na trake, eko nam ovi podaci ikada ponovo zatrebaju, vratićemo ih ne disk i tamo ih obraditi. Traka je, dakle, postelja tipično sekvencijalni medij: podaci se upisuju od početka prema kraju. Ukoliko pleniraate da koristite magnetnu traku ne ovej našoj, na trebe de bacate novac ne kupovinu skupih jedinica: prave su stver takozvane strimer trake (jedne standardne jedinice magnetne treka obično košte koliko i 3—4 strimer jedinice) koje nemeju precizan mehanizam za zaustavljanje ali koje omogućavaju brzo sekvencijalno upisivanje i čitanje podataka.

Do sada smo govorili isključivo o velikim kompjuterskim sistemima i to se razlogom — ne njima su se razvijale metode za sortiranje i pretraživanje podataka. Kakva je situacija na spektrumima, komodorima i drugim kućnim mašineme? Mnogo složenije.

KAKO UČITATI PODATKE

Pripremajući program sa slike 1 na standardnom paskalu susreli smo se sa jednim problemom koji će, verujemo, poneked pogoditi i vas. O čemu se radi?

Traća de učitemo dve serije podataka nizove A i B. Podatke upisujemo jedan po jeden i tako dolazimo do kraja niza A. Kako de signaliziramo da u A nema više podataka? Najjednostevnije je da pritisnemo CTRL Z ili neku drugu sličnu kombinaciju tastera i teko stavimo računeru do znanja da je fajl koji smo unosili završen. Promenljiva **eof** *će tada dobiti vrednost* **true**, *funkcije* **readeer** *će se zevršiti i kontrole će se vratiti glavnom programu.*

Nevolje je što sada traba de učitemo i drugi niz. Što se računere tiče, „detoteke **input** *koja je dodeljene tasteturi je pročitene — u njoj neme više podataka tako de sledeće* **readin** *jednostavno neće fukcionisati. Šta de se sa radi? Prirodno bi bilo iskoristiti naredbu* **rewind** *kojom se sekvencijalna detoteka koja je (evetualno) isčitana do kraja, premoteve ne početak. Ne želost, standerdi ISO Pascala ne određuju kako* **rewind** *deluje na detoteku* **input** *— u knjizi eksplicitno piše da ovo delovenje može de bude zavisno od implementacije. Autori implementacije koju mi konstimo su sabi oleksali posao pe su učinili da* **rewind** *uopšte ne deluje ne detoteku* **input** *što je, verujemo, sasvim klasičan slučaj.*

Šta de se radi? Oformili smo pomoćnu datoteku **douminp** *koja je definisana kao* **text** *(ili šta je isto,* **file of char**), *a zatim je dodelili testaturi. Ovu detoteku možemo jednostavno da „premotavamo i de čitamo nekoliko puta unoseći svaki put nove nizove. Podaci se, dakle, mogu komforno unositi i bez standardne detoteke* **input**!

Personelni računeri često koriste standerdni kesetofon za skladištenje podateke. Standerdni kasetofon ne trebe poreditи čak ni se strimer trekom: mnogo je sporiji, podložniji greškame i što je najvažnije, na omogućave kompjuteru **nikakvu** kontrolu premotevenje. Kasete je, dakle, sekvencijalni medij koji se kraće isključivo od početka prame kreju: bilo kakvo premotavenje zavisi od korisnike koga ne treba previše opterećivati. Zbog ovih ograničenja *nije moguće sortirati veće količine podataka upisanih na kasetu!*

Razočarali smo vas! Dobro, dejte nam još jednu šansu: podatke upisene ne treku možete de sortirate eko eu vem pri ruci dve (ili još bolje tri) kasetofone. Opet sta razočarani: nemate tri kasetofona, ipak ne želoetita se baš mnogo i de ih imate: podatke biete mogli de sortirate tek ukoliko vaš računer može **istovremeno** da kontroliše (uključuje i isključuje) **eve** tri uređaje. A to koliko nam je poznato, ne može ni jeden populeran personalni kompjuter!

Nećemo se, međutim, tak teko oprostiti od sortirenje podateke na trekame: neka od metoda koje ćemo upoznati će nam koristiti dacnije. Kako ćemo dakle, sortirati podatka upisene na kasetu ukoliko posedujemo tri kasetofone koje računer kontroliše? Podimo redom i upoznajmo za programiranje neobično važen pojem *mešanja* podateka.

Mešanje . .

Deta eu nam dve deloteka koje ćemo nazvati A i B: sveka od njih sadrži već sortirene podatke (npr. A =(1 3 7 12 123) i B = (−4, 0, 10 12, 47). Trabe de proizvedemo detoteku C koja sedrži sve podatke iz detoteka A i B, pri čemu će i C biti sortirene u našem je slučaju C – (−4 0, 1 3, 7 10 12 12 47, 123). Do rezultata se dolazi sasvim jednostavno što pokazuje i paskal program se slike 1.

Uzmemo jeden elemenat iz detotake A (1), e onda i elemenat iz B (−4). Uporedimo ih pe manji (−4) upišemo u C a onde uzmemo novi elemenat iz B. Radi se o nuli koja je i dalje manje od 1 nula dakle putuje u C. Siedeći elemenat iz B je 3, broj veći od nule. Zeto nulu šeljemo u C i uzimemo novi elemenat iz A (7) koga poredimo sa 3. Postupek poredenje i uzimanje sledećeg elementa iz detoteke čiji je član bio manji ponavljemo sve dok se jedna od detoteke (u našem slučeju B) ne isprazni tede ostetak druge detoteke jednostavno prepišemo u C.

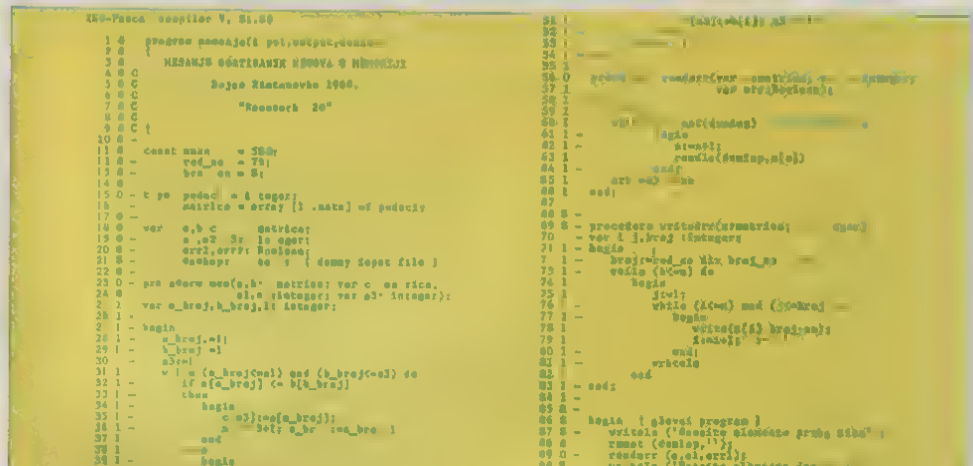
Rekli bi ee de je postupak jednostavan i logičan i de o njemu name smisla govoriti. Jeste li, međutim, primetili da je mešanje se slike 1 sekvencijalno: de A i B nisu nizovi već detoteke ne spoljnim medijime. C bismo i dalje mogli de formiramo učitavajući jeden po jeden element iz A i B bilo koje od ove tri detoteke može de bude mnogo veća od memorije našeg računera!

. . i drugi trikovi

Vratimo se prvobitnom problemu: trabe sortirati podatke iz detoteke A ne treći koristeći tri kesetofone. Posao obevljamo u nekoliko faza.

1. Računer učitave što veći segment se treke 1 u memoriju i sortire ge primenom neke od klasičnih metode (npr. heap sort ili

U prošlim smo „Računarima", govoreći o sortiranju brojeva, objavili nekoliko „klasičnih" rešenja i uporedili njihove karakteristike. Zaključci do kojih smo došli, međutim, važe samo ako su podaci koje sortiramo upisani u memoriju našeg kompjutera. Rad sa velikim bazama podataka koje su smeštene na diskovima, disketama i trakama nameće nova pravila igre.



```
ISO-Pascal compiler V, 81.80

  1 0   program memorija(i pot,output,demo
  2 0   {
  3 0      MESANJE SORTIRANIH NIZOVA U MEMORIJI
  4 0 C
  5 0 C           Dojan Kintanovko 1988.
  6 0 C
  7 0 C              "Kosovsk  20"
  8 0 C
  9 0 C }
 10 0 -
 11 0   const max    = 300;
 12 0 -       red_no  = 79;
 13 0 -       bro  on = 8;
 14 0
 15 0 - type  podac  = integer;
 16   -       matrice = array [1 .max] of podaci;
 17 0 -
 18 0   var   a,b c       matrice;
 19 0 -       a ,a2  3:  integer;
 20 0 -       err1,err2: Boolean;
 21 0 -       demop:     ne  ;  { dummy input file }
 22 0 -
 23 0 - procedure mem(a,b:  matrice; var c  matrice,
 24 0                 a1,a  :integer; var a3: integer);
 25 1   var a_broj,b_broj,1: integer;
 26 1 -
 27 1 - begin
 28 1 -    a_broj:=1;
 29 1 -    b_broj:=1
 30   -    a3:=1
 31 1      while (a_broj<=a1) and (b_broj<=a3) do
 32 1 -       if a[a_broj] <= b[b_broj]
 33 1 -       then
 34 1 -          begin
 35 1 -             c[a3]:=a[a_broj];
 36 1 -             a[a3]:=a[a_broj]; a_br  :=a_bro  1
 37 1             end
 38 1          else
 39 1 -          begin
 40 1             a[a3]:=b[b_broj];
 41 1             a3:=a3+1; b_broj:=b_broj+1
 42 1 -          end;
 43 1 -       petlja se eventualno i vrsava 0 puta! }
 44 1         for b:=a_broj to a1 do
 45             begin
 46   -             a[ 3]:=a[ i]; a3:=a3+1
 47 1 -          end;
 48 1         petlja se eventualno i vrsava 0 puta!
 49 1         for  :=b_bro  to a.
 50   -          be  e
 51 1                 [a3]:=b[i]; a3 :=
 52 1 -
 53 1          }
 54 1 -
 55 1
 56 0   procedure  readerr(var  matrice; var  integer;
 57 1                     var err:Boolean);
 58 1
 59 1
 60 1      var          ant(demop)
 61 1 -          begin
 62 1 -             a:=a+1;
 63 1               readln(demop,a[a])
 64 1 -          end;
 65 1      err:=a1 ...
 66 1   end;
 67
 68 0 -
 69 0 - procedure writedrr(a:matrice;       ...)
 70   - var i j,broj :integer;
 71 1 - begin
 72 1 -    broj:=red_no div broj_no
 73 1 -    while (b<=a) do
 74 1         begin
 75 1            j:=1;
 76   -          while (i<=a) and (j<=broj
 77 1 -             begin
 78 1                write(a[i] broj:ne);
 79 1                i:=i+1;
 80 1 -             end;
 81 1 -          writeln
 82 1         end
 83 1 - end;
 84 1 -
 85 0 -
 86 0   begin  { glavni program }
 87 0 -    writeln ('Unesite elemente prvog niza' );
 88 0      reset (demop,'');
 89 0 -    readerr (a,a1,err1);
 90 0      writeln ('Unesite elemente dru ...
 91 0 -    reset (demop,'');
 92 0      readerr (b,a2,err2);
 93 0 -    if   err1 or err2
 94         then writeln ('Pravlas podataka! )
 95 0 -    else begin
 96                mem (a,b a,a1,a2,a3);
 97 0              writeerr (c,a3)
 98   -          end
 99 0 - end.
0 Compilation error(s)
Code size = 979 bytes
```

Stika 1

particijski sort) Ovi se podaci upisuju na traku 3

2 Računar učitava sledeći segment podataka sortira ga i upisuje na traku 2 Faza 2 se ponavlja sve dok ima podataka, tako su na traci 2 formirane datoteke A1 A2 A3 An-1 jednakih veličina i An koja je eventualno manja

3 Računar zahteva od korisnika da premota trake 2 i 3 na početak i da u kasetofon 1 umetne novu, preznu kasetu

4 Računar meša podatke sa traka 2 (A1) i 3 i formira novu datoteku na traci 1

5 Od korisnika se zahteva da premota traka 1 i 3 na početak i da im zatim razmeni mesta Koraci 4 i 5 se ponavljaju sve dok se ne promešaju sve datoteke sa trake 2

6 Sortirani podaci su na traci 1 koju još treba premotati na početak da bi se rezultati ispisali na ekranu ili upotrebili na neki drugi način

Iako je čitav postupak prilično komptikovan smatrali smo da se vredi potruditi i napisati program 2 koji simulira čitavu operaciju

Program je pisan na standardnom bejziku i u njemu nećete naiči ni na kakvo pominjanje datoteka Za to smo imati nekoliko dobrih razloga Pre svega, teško je testirati program koji se stalno obraća sporom kasetofonu Dalje rad sa datotekama na traci nije baš standardno rešen u raznim bejzicima Treće (i najvažnije) autor ovoga teksta bi još negde i pronašao kompjutarski kasetofon ali bi pronalaženje kablova bilo pravi problem

Uveli smo, dakle, matricu *traka* koja simulira kasete *traka (0 5)* je peti element na prvoj traci Na prvu su traku u početku upisani slučajni brojevi koje treba sortirati na istoj će se traci na kraju naći i rezultat Tu je naravno, i niz *memorija* koji je dimenzionisan tako da bude mnogo manji od svaka od „traka Računar će, kada mu god zatreba ispisati uputstva operatoru u stilu „razmeni traka u kasetofonima 1 i 3" ili „premotaj traku u kasetofonu i a zatim će ovu operaciju samostalno izvesti — treba samo da primetite koliko bi u realnosti bilo posla

Testiranje je pokazalo da je program sa slike 2 prilično spor na inače brzom BBC-jevom bejziku sortiranje 300 brojeva zahteva 3 minute rada da se računar stvarno obraćao kasetofonu posao bi trajao mnogo duže Može li se nešto ubrzati? Bilo bi pametnije mešati datoteke A1 i A2, A3 i A4, itd, a zatim dva po dva rezultate Teda bi, međutim, trebalo mnogo više pomoćnih kaseta i žongliranja sa njima

Pre nego što se (zauvek) oprostimo od kasetofona pomenućemo još jedan veoma ozbiljan problem Čak i kada bi spektrum mogao da kontritiše tri kasetofona postupak koji smo opisali se na njemu ne bi mogao primeniti jer je zapis podataka na traci kontinualan ne možemo da učitamo 100 bajtova da zaustavimo traku i da docnije učitavamo dalje Zašto ne možemo? Čak i kada bi traka mogla da se zaustavi i pokrene bez „preznog hoda koji je izazvan inercijom računar neće pronaći novi tider pa će ignorisati ostatak podataka! Mnogo je zgodnije ako računar organizuje podatke na traci po blokovima pa se svaki blok može nezavisno učitati — toliko što se tiče kritika pojedinih stručnjaka koji smatraju da je sackanje programe na blokove loša strana amstrada BBC-ja i elektrona !

Pravo u glavu

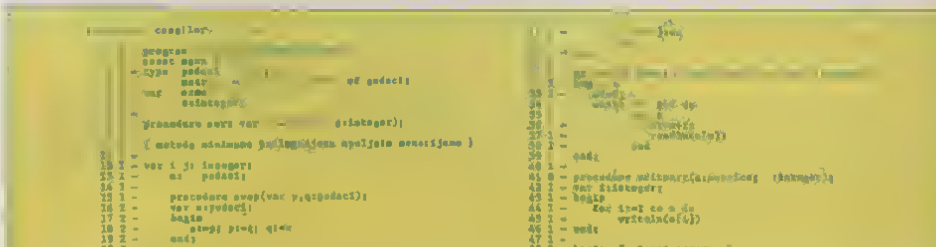
Često se kaže da je disk jedinica uređaj koji omogućava direktan pristup podacima. Šta beše direktan pristup? Ako su podaci upisani u RAM ili ROM vašeg kompjutera, možete potpuno slobodno da pristupate bilo kom od njih pošto ste pročitali sadržaj ćelije &A000A i &C000C je potpuno jednako

```
 10 REM
 20 REM  Simulacija sortiranja podataka
 30 REM  na trakama primenom 8 kanalska
 40 REM
 50 REM        Dejan Ristanović 1986
 60 REM
 70 REM             "Racunarf  20"
 80 REM
 90 REM
100 ON ERROR GOTO 1000
110 casi=300   : REM cas  gasenika za iracl
120 cuse=20    : REM cas  gasenika u memoriji
130 cuxgas=300: REM kolika se potrosnja darlirk
140 DIM treka(2,casi) tcasi(2)
150 DIM cas(cuse)
160 REM
170 REM Program podataka za iracl 8
180 REM
190 FOR i=1 TO cuxpad
200    treka(0 i)=RND(1000)-500
210 NEXT i
220 treka(0,maxpad+1)=endfile
230 REM
240 REM Poduci su treka 0 za rukhijnje
250 REM na duzeve valicina  cuse' koji
260 REM se sortiraju i upisuju na
270 REM trakal i treka2
280 REM
290 PRINT "Stavi traka za podacima n kasetafon 1"
300 PRINT "Stari gtasnu traku n kasetafon 2"
310 PRINT "Sim•i pravcu traku u kasetafon 3"
320 PRINT
330 eof=FALSE
340 dBK i=0 TO 2:tcuvnt(i)=1:NEXT i
350 ROH
360 cuse=2
370 cas_cnant=1
380 iepe_na=0:GOSUB 990
390 I9 eof THEN 420
400 cas(cas_tcasi)=kyte_s
410 cas_tcasc=cas_tcasi+1
420 I9 cas_cunst<=cuse THEN GOTO 280
430 s=cas_tcunt-1:IF s=0 THEN GOTO 460
440 GOSUB 1130: REM Sort
450 FOR i=1 TO cas_cnant-1
460 iepe_as=pozai:kyis_s=cas(i) GOSUR 1070
470 NEXT i
480 tepe_su=pozai:kyis_s=endfile:GOSUB 1070
490 IF NOT eof THEN pra=s+1:GOTO 370
500 tepe_as=pozai:kyte_s=endfile:BOSUB 1070
510 REM
520 REM Nosnaje podataka as trakal
530 REM i traka2  Zasalisi se icake8
540 REM
550 PRINT "Prasnisaj trakn u kas  2 na pocatak":tcuvci(1)=1
560 PRINT "Prasutaj traku o kas  5 as gecatak":tcunt(2)=1
570 PRINT "Sturi grasnn iraka n kasnietafon 1" casi(0)=1
580 PRINT
590 eof=FALSE
600 tepe_as=2:GOSUB 990:I4 eof THER 660
610 gas@yte_t
620 tepe_as=1:GOSUB 990:IF eof THER 700
630 bkykte_r
640 I9 as<>b THEN 650
650 kyie_s=catiepe_ns=0:GOSUR 1070
660 iepe_ns=2 GESUE 990 IF eof THEN 860
670 as=kyte_r:GOTO 640
680 kyis_s=kkrtepe_ns=0:GOSUP 1070
690 GOTO 620
700 kyie_s=as:iepe_as=0:GOSR8 1070
710 eof=FALSE
720 tepe_as=2:GOSUP 990:IF eof THER 750
730 kyie_s=kyte_ritepe_ns=0:GOSUB 1070
740 GOTO 720
750 iepe_as=0:kyte_s=endfile:GOSUB 1070
760 IF iraka(1,tcuvnt(1))=endkata THEN 910
770 REM
780 REM  Nosnsne traka i priprnse sa
790 REM      sliedeca sesnaaja
800 REM
810 PRINT "Rasnsat iraka n kasnsafsnisa 1 i 3"
820 PRINT "i oka prasntaj na pacelak "
830 PRINT
840 FOR i=1 TO cas:tiraka(2 i)=traka(0 i) NEXT i:tcuvci(2)=1 cunst(0)=1
850 GOTO 590
860 iepe_as=0:kyie_s=kk GOSUR 1070
870 eof=FALSE
880 iepe_as=1:BOSUR 990:IF eof THEN 750
890 kyie_s=kyte_s:iepe_as=0:BOSUB 1070
900 GOTO 880
910 PRINT:PRINT:PRINT "Sortiraaja je cartcesn "
920 PRINT "Prasnisaj iraka i na pratjek "tcunt(0)=1
930 PRINT
940 eof=FALSE:iepe_as=0
950 GOSUB 990:I9 NOT eof THEN PRINT kyte_r;:GOTO 950
960 PRINT:PRINT
970 END
980 REM
990 REM Bajt za iraka 'kepe_as'
1000 REM    ocitaiS s 'kyte_r'
1010 REM
1020 kyie_r=kraka(kepe_as,cuunt(kepe_as))
1030 cuvci(kepe_as)=cuvci(kepe_as)+1
1040 IF kyte_r=endfile THEN eof=TRUE
1050 RETUSN
1060 REM
1070 REM Bajt 'kyie_s' aplasii
1080 REM    as traka 'iepe_as'
1090 REM
1100 traka(iepe_as,cuvnt(tepe_as))=kyie_s
1110 cuvnt(tepe_as)=cunt(tepe_as)+1
1120 RETURN
1130 REM  BRAP SORT
1140 FOR k=2 TO n
1150 h=k
1160 GOSUB 1250:REM UP
1170 NEXT i
1180 FOR i=n TO 2 STEP -1
1190 s=cas(1):cas(1)=cas(i):cas(i)=s
1200 h=1-1
1210 GOSUB 1520:REM BOWN
1220 NEXT i
1230 RETURN
1240 RSM
1250 REM fasciisa UP
1260 REM
1270 l=h
1280 IF l<=1 THEN RETURN
1290 p=INT(l/2)
1300 IF cas(p)>=cas(l) THEN RETURN
1310 s=cas(p):cas(p)=cas(l):cas(l)=s
1320 l=p
1330 GOTO 1260
1340 REM
1350 REM fasciisa DOWN
1360 REM
1370 i=1
1380 p=2*i
1390 IF p>h THEN RETURN
1400 IF p<h AND cas(p+1)>cas(p) THEN p=p+1
1410 IF cas(i)>=cas(p) TH N RETURN
1420 s=cas(p):cas(p)=cas(i):cas(i)=s
1430 i=p
 1440 GOTO 1380

>RUN
Stari itaka as podarius n kasntafoa 1
Stari grasnu traku n kasata•an 2
Stari prasan traku n kasnsnion 3

Prasntaj iraku n kas  2 na pocatak
Prasntaj iraku n kas  3 as garatak
Stari prasan trakn v kasntsius 1

Rarsnni iraka s kasntofsalns 1 i 3
i nka prasntaj na pocatak

Rarsnni traka  kasntsisnina 1 i 3
i nka prasnsinj na pscatak

Rasnni traka n kasntsisnina 1 i 3
i obu prasntaj na garatak

Rarsnni traka n kasntsisnina 1 i 2
i nka prasntaj na porstak

Rarsnni traka s kasntofsni s 1 i 3
i nka prasntaj as pocatak

Rasnni traka s kasntsisnina 1 i 3
i nka prasntaj na porstak

Rannni traka n kasntsionina 1 i 3
i nka prasnsinj is pscatak

Rarsnni iraka n kasntafsnina 1 i 3
i n•a prasntaj na pocatak

Rarsnni traka  kasntofsnina 1 i 3
i obu prasntaj as garatak

Rarsnni traka s kasntafsntsn 1 i 3
i nka prasntaj as pocatak

Rarsnni iraka s kasntafsnina 1 i 3
i nka grasntaj na pocatak

Rasnni traka v kasntafsnina 1 i 3
i nka prasntaj as garatak

Rarsnni traka s kasnsinfsntsa 1 i 3
i nka prasntaj na gocatak

Sortiraaja je cartcesn.
Prasntaj traka 1 as pocatak.
```

slika 2

Ukoliko sa druge strane pročitate prvi bajt sa traka čitanje sledećeg ćete moći da obavite praktično trenutno dok ća pristup bilo kojem bajtu potrajati. Treka ja dakle medij sa *sekvencijalnim* pristupom dok memorija računare predstavlja medij sa *direktnim pristupom* (pod memorijom podrazumevamo ROM i RAM — i jedan i drugi omogućavaju slobodan pristup podacima premda je samo RAM (nesrećno izabrana) skraćenica od *Random Access Memory)*

Disk ja, sa strane korisnika takoda medij sa direktnim pristupom ako smo pristupili osmom bajtu sa treća trake nema prepreka da odmah zatim pristupimo trećem bajtu osma traka. Poznavanje karakteristika diska sa druga strane opovrgava opšte poznata tvrdnja izbor trake obavljaju dekodarski elamenti koji zaista deluju praktično trenutno. Kada, međutim, dekoder izda naredbu upisno-čitajućoj glavi da putuje do osme traka na scanu atupa mehanika glava će brže stići do osme nego do osamdesete treke! Pokažimo ipak malo dobre volje i zanemarimo vrama putovanje glava. Kada glava dođe do osme trake treba još da pronađe treći bajt. Diskata se okreće određenom brzinom i to isključivo u jednom smeru — ukoliko , glava ima sreće treći će bajt naići upravo kada se ona nađe iznad osme traka pa će učitavanje biti praktično trenutno. Ukoliko na glavu deluju Merfijevi zakoni treći ja



stika 3

bajt upravo promakao pa će za njegovo učitavanja diskete morati da napravi čitav krug. Možemo dakle da smatramo da će u nekom prosečnom atučaju diskete napraviti pola okretaja pre nego što se pročita željeni bajt. Dok se ova rotacija obavlje disk kontroler čaka bajt a računar čeka disk kontroler — sa auštinska strana, situacija je ista kao da smo zahtevali učitavanja hiljaditog bajta kasete koja je pramotana na početak. Disk jedinica, dakle, kombinuje diraktan i sekvencijalan pristup podacima (diraktno se pristupa tragu a sekvencijalno sektoru ili bajtu) ali se zahvaljujući brzini njenog okretanja **u nekim apitkacijama** može uspešno aproksimirati uređajem sa diraktnim pristupom.

Sortiranje, na žalost nije jedna od aptikacija u kojoj smamo da se uljuljkujemo u verovanje da se podecima na disketama slobodno pristupa. Moraćemo, šta više, da se upustimo u dalja proučavanja komunikacije računer — disk i upoznamo pojam batera.

Računar ja da ostanemo kod ranijeg primera tražio učitavanje trećeg bajta osmog sektora diskete. Glave je došla do osma trake ali — kako da pronađa treći bajt? Tu ja naravno apsenonani otvor koji algnetizira početak traka i u odnosu na koga treba brojati bajtove i pored praciznu izrada mehanika nemoguće ja biti alguran koji ja bajt koji a traba i predvidati naku vrstu ćeksuma koji će proveravati ispravnost prenosa podataka. Ćek i ukoliko zanemerimo ove probleme, obeshrabriće nas jedan paredoks pratpostavimo da ja računar primio treći bajt i da mu ja odmah zatrabao stedeći, četvrti. Dok ja odgovarajući zahtev poslat disk kontroleru diskete se u nepraktdnom okretanju pomerila i bajt je „promakao" — da bi mu se pristupilo, treba sačekati da diskata napravi čitav krug. Ukoliko dakle u memoriju učitavamo program koji uvek predstavlja niz aukcesivnih bajtova disketa će morati da se okrene onoliko puta koliko program ima bajtova — to bi trajalo totiko dugo da bi se svi vratili trakama pa bi jadni proizvođači disk jedinica ostali baz nseušnog hleba. Rašenja je dakle, moralo da sa pronađe pa ja i pronađeno uvođenje batera rašava problam učitavanja aukcesivnih bajtova problem adresiranja bajtova i problam čakaumal. Da vidimo kako.

Svaka je traka podeljena na odreden broj saktora na primar njih 10. Svaki sektor se druge strana sedrži odredeni broj bajtova, na primer 256. Umesto da pristupamo tračem bajtu osme traka pristupečamo dekla, tračem bajtu **nuttag saktora** osme traka dok ćemo pristup 2073 bajtu 12 traka zamenlti pristupom dvadeset patom bajtu **aamog sektara** dvenaeata traka. Adresiranje amo dakle iskomplikovali, ali smo ponešto i dobili. Pra svega svaki sektor *može* asbi da dopusti luksuz da oalm korisnog sadržaja, obuhvati identifikaciju (broj sektora) i odgovarajući ćeksum. U *memoriju* računara se neće pranositi samo jeden bajt već čitav sektor ukoliko smo tražili 2073 bajt dobićemo uz njaga i 2048 2049, 2050 itd — posledni će biti 2303 bajt.

Problem adresiranja sektora i provere ispravnosti prenosa amo dakla rašili. Kako atoji atvar sa sukcesivnim bajtovima? Vrlo lapo ako nam posla 2073, zatrabe 2074 bajt nećemo moreti ćak ni da sa obraćamo disku. bajt ja u bateru (tj. u RAM-u) pa mu pristupamo praktično tranutno. Ukoliko nam budu trabali i detji bajtovi disk će moći da miruja sve dok ldući redom na zatražimo 2304 bajt kada će se, jasno ponovo prepiseti celih 256 bajta.

Šta se menja ako, umesto prostog čitanja, želimo da menjamo podatke na disku? Kada zatražimo promenu 2073 bajta dvanaesta traka u batar će biti prapisano 256 bajta, a onda će izmana biti upisana u RAM ali **ne** i na disk. Dok budemo menjali bajtove čije su adrase izmedu 2048 i 2303 operacija će se obavljati u RAM-u i biće izuzetno brze, kada pristupimo 2304 bajtu čitav će se batar prapiseti na disk a onda će u memoriju biti učitan stadaći sektor od 256 bajta apraman za izmana. Ukoliko dakla u ared rada nestane struje, naki promanjani podeci neće biti upisani na diskatu pa sadržaj datoteke neće biti sasvim ažuran što svakako nija prednost. Radi se ipak o maloj ceni koje ja plačena za atrahovito ubrzavenja rada.

S kraja na kraj

Obzirom da ja pristup RAM-u praktično tranuten a pristup disku apor sortiranje će biti strahovito aporo ako, koristeći naki inače dobar matod nateramo glavu da šeta se kraja na kraj datoteka i da za svaki potraban bajt učitave i upisuja njih 256. Pra nego što kažemo kako se podaci najčešće sortiraju, podsetićemo se matoda koje amo upoznali u prošlom broju i pogledati ih u novom svetlu.

Zdravorazumeka *matoda minimuma*, začudo, nije loša, ali je nikako na emate primanjivati kao što smo opisali u prošlom napiau. Na slici 3 dajemo, zato, novu verijantu programa.

U čemu ja razlika? Sasvim aitna: umesto da poredimo A(I) i A(J), poredimo M i A(J) pri čemu amo promanljivoj M najpra dodelli vrednost A(I). Zašto uvodimo pomoćnu promenljivu? Zamislimo da je I=20 J=400. Pristupamo najpra dvadesetom bajtu pa, dakle, učitavamo čitavih 256 bajta. Zatim pristupamo 400-tom, bajtu zbog čega moramo da prapišemo drugih 256 bajta u bafer pre čega amo prvih 256 vratili na diskatu. Zatim je J postalo 401 i treba da poredimo A(20) i A(40). Da bi računar saznao koliko je A(20) mora ponovo da učita prvi saktor i da onde čita drugi da bi saznao koliko ja A(401). Ukoliko sa druge strane, A(20) pranesamo u promenljivu M koja se normetno čuva u RAM-u iz datoteka ćemo čitati sukcesivna A(J) i tako konstiti sve blagodeti batera.

Matoda mahurića je možda još pogodnija i to baz ikakvih izmana one i tako podrazumeva sukcesivno pristupanje brojevima koji se sortiraju. Odradani problem nastupa kada pratupamo bajtovima na granicame sektora (ponakad ih traba i razmeniti što podrazumeva bar četiri učitavanja sektora) ali — to ja jeden bajt na 256. Ukoliko volite izazove pokušajte da modifikujete program iz prošlih ,Računara tako da minimizirate ovaj problam.

Šelova sortirka nije neročito pogodna — česti pralazi prako granice sektora dovode do napotrabnog prenošenja podataka na relaciji računar — diak.

Hasp sort nija mnogo botji III, da budemo meto praciznji, nije bolji u osnovnoj verziji. Pra samo godinu dana američki časopis *ICK Journal* objavljuje modifikovan algoritam za *heap sort* koji je prilegođen operativnim sistamima IBM-ovih računara iz serije 370 ali verujamo da ovaj algoritam na bi bio posebno interesantan za večinu naših čitatača.

Quick sort naizmenično pristupa levom desnom kraju niza, što znači da ja katastrofalno loš za klasično organizovene bafere. Poneki operativni sistem se druga strane omogućava da se jednoj datotaci dodeli nakoliko bafera, u kom bi slučaju *quick sort* mogao da buda interesantne alternative klasičnim matodama.

Da vidimo najzad kako se atvarno sortiraju podaci na disku. Ne leti način kao podaci na traci mešanjem. Učitavamo dakle što veće segmente podataka u RAM (ne samo u bafer nago u čitav slobodan RAM pa ih sortiramo nakom od brzih metoda

(npr. particijski). Ukoliko je datoteka imala M podataka, a u našu memoriju može da stane njih N, ovim ćemo postupkom dobiti otprilike M/N potpuno sortiranih datoteka, koje treba ujediniti u jednu. Mešaćemo, dakle, dve po dve od M/N datoteka i tako korak po korak formirati konačnu. Vrlo je zgodno da, ako posedujemo dve disk jedinice, ne smestimo sve tri datoteke na istu disketu — smanjićemo nepotrebno „šetanje" glave. Najbrže bi bilo da imamo tri disk jedinice, pa da na dve smestimo datoteka koje se mešaju, a na treću rezultat, ovakav luksuz, ipak, retko ko sebi dopušta!

Klasična kompjuterska literatura, kao što je knjiga *Organizacija podataka* Hartmuta Vedekinda, opisuje i mnoge druge metode sortiranja podataka na trekama i na diskovima, pomenućemo simetrično i asimetrično mešenje, kaskadno sortiranje i oscilirajuće sortiranje. Obzirom da su magnetne trake u mnogome stvar prošlosti, nećemo se posebno baviti ovim metodama, ali ćemo zato, kada se budemo ozbiljnije pozabavili bazama podataka, upoznati i jedan sasvim drugačiji metod za sortiranje podataka, formiranje takozvanih indekas, koje podrazumeva da se podaci koja sortiramo uopšte ne pomeraju sa mesta!

Obrtanje u prazno

Počeli smo od trake, govorili o flopi disku, pa je red da posvetimo nekoliko reči danas sve popularnijim i jeftinijim masivnim diskovima. Hard disk se, za razliku od diskete, neprekidno okreće bez obzira na to da li su vam trenutno potrebni neki podaci ili nisu. Kada ispred računara sedi jedan korisnik, ovo je okretanje prilično neracionalno, korisnik se, bar pri upotrebi većina aplikacionih programa, razmerno retko obraća disku. Ukoliko se, sa druge strane, radi o sistemu koji opslužuje gomilu korisnika, disk će biti veoma zaposlen, na samo da su se zahtevi korisnika, poređali u red, već i sam operativni sistem, zbog smene stranica i drugih „kućnih poslova", zahteva česte pristupe disku. Ovim razmatranjem nemamo, naravno, nameru da kažemo da je hard disk neisplativ za pojedinca — kažemo samo da je iskorišćenost diska u vremenu mala.

Sortiranje podataka na masivnom disku se, suštinski, ne razlikuje od sortiranja podataka na disketi, sve se operacije jednostavno brže obavljaju! Ekvivalentnost medija je dopunjena time što većina operativnih sistema na personalnim računarima tretira hard disk kao i flopi — svekoj sa datoteci dodeljuje po jedan bafer konstantne veličine. Ukoliko vaš kompjuter ima moderan operativni sistem poput „amiginog", možda će se isplatiti upotreba višestrukog baferisanja i optimizacija veličina bafera. Ukoliko, sa druge strane, radite sa XT-om ili nekim kompatibilcem koge ste opremili hard diskom, zadovoljite se napomenama koje smo dali korisnicima disketa.

Naučili smo, dakle, da sortiramo podatke u memoriji i podatke na diskovima, ostalo je još da te podatke obrađujemo, tj. pretražujemo. A o pretraživanju podataka — u sledećim „Računarima".

Dejan Ristanović

Računari u domaćoj radinosti

Turbodrajv za „spektrum"

parče mašinskog koda

Poslednjim bajtovima mašinskog koda operativnog sistema za rad sa diskom, turbodosa, koji objavljujemo u ovom broju, primakao za kraju i drugi naš ovogodišnji veliki projekat za vizanike još uvek vitalnog „spektruma". Preostaje još samo program za unošenje mašinskog koda, — za sledeći broj pripremamo jedan izuzetan komforan heks-loader, mada se sasvim lepo može koristiti i loader koji je objavljen uz „Ekranski editor".

```
33376 00 00 00 00 00 00 00 FF FF CD 1C 0B B5 DD 7B 19 CD 18
33392 7B 09 D8 E5 E1 11 52 00 19 E5 01 01 00 CD 21 0A 7F
33408 C8 BA 0B 30 04 BB CB 18 B6 A7 20 02 B7 05 DD 6B DF
33424 1A 85 66 1B F5 01 02 00 CD 21 0A C1 CD 70 29 B1 B0
33440 BB CB 18 A6 20 05 CD A8 0D 1B 23 B5 3E 2E 90 47 15
33456 11 0B 00 19 1D FD AF 77 06 0A DD 85 23 DD 7E 0E C6
33472 77 DD 23 10 F7 DD E1 E1 01 01 00 CD 60 0A B1 C9 00
33488 83 7B 19 CD 7B 09 DD E5 E1 11 52 00 19 C0 0A 0D 95
33504 CD 21 0A 13 1A DD 77 45 13 1A B6 C0 FE B0 38 6A
33520 02 3E B0 DD CB A3 CE 20 04 DD CB A3 BE 07 B0 01
33536 77 46 DD 7E 0D DD 77 44 A7 C9 E5 0D 66 1B BD 6B B8
33552 1A 7E 32 CF 5C E5 CD 60 0B E1 3E 55 23 7E 2B FE AD
33568 3F 7E 20 03 3C 1B 02 3E 02 4F 3A CF 5C B9 20 0E 0E
33584 B8 CB 28 7E 20 02 E7 09 A7 E1 C9 06 00 C5 4E DD AB
33600 7E 19 CD 7B 09 C5 01 7F A0 ED B0 C1 06 CO ED 79 D6 01
33616 00 01 7F 00 3E 10 8D 79 CD 9A 0A ED 7B 1F 3B F8 56
33632 CB 6F C1 20 03 CD 60 0A C1 CD 21 0A 1B A7 81 18 66
33648 DB C5 7E 3D A7 0E 11 3E 01 10 F3 C1 AF 3F A9 90 B9
33664 3D 06 0A 90 3E 03 0C 18 FA B0 3C 41 4F E1 37 C9 63
33680 06 AA 23 23 7E DD CB 29 7E 20 1B E6 3F DD 8E 29 E4
33696 23 20 06 7E DD 86 00 2B 06 23 23 10 E7 A7 C9 37 81
33712 E5 D1 C9 FE 3F 23 28 F7 1B EF B5 BD E5 AF DD 77 9F
33728 36 11 0B 00 19 0E 20 06 0A 7E 23 FE 3F DD E1 DD 2F
33744 B5 28 11 00 7E 0E BB 20 08 23 DD 23 10 F5 37 79 4B
33760 00 E1 01 C9 DD 71 36 58 19 00 20 DD 4F 0D E1 B0 9F
33776 7E 36 18 EE 3A 06 5C C0 78 09 C0 9F 10 24 E9 5C 5F
33792 22 E4 5C CD 3B 0D 0D CA 1B 46 20 05 C0 EA 0F E7 4C
33808 0C DD C8 43 D6 DD 7E 19 CD 78 09 DD B5 E1 11 52 95
33824 00 19 E8 21 E6 5C 01 09 00 DD 71 08 E0 B0 05 21 5D
33840 09 00 E0 4B E7 5C 09 C8 3C 24 D1 2A E4 5C E0 4B 2B
33856 87 5C 7B E1 2B 22 DD 78 0C FE 02 20 0F E5 C5 CD C3
33872 E0 10 C1 D8 E5 E1 11 52 00 19 EB E1 ED 40 DD 34 3A
33888 0D 20 DF DD 34 0C 1B D4 DE C8 43 CE CD E0 10 C3 52
33904 F8 0F 2A E1 5C 22 E4 5C CB 3B 0D DB C8 1B 46 28 15
33920 05 CB 2E 11 E7 11 11 DD CB 43 56 1B 02 E7 16 DD 4F
33936 E5 E1 11 52 00 19 11 E6 5C 01 09 00 ED B0 C9 22 27
33952 B9 5C DD 5E 53 00 56 54 21 0B 00 19 CB 3C 24 7C 43
33968 32 E7 5C B0 34 0D 11 09 00 DB 6B A5 DB 66 A6 B7 7D
33984 ED 52 D0 45 3D 74 46 ED B5 E1 11 5B 00 19 EB 02 82
34000 5B E9 5C 19 14 C0 40 00 1B 7B 4A B0 34 0D 17 3D 5D
34016 57 0F 1B F7 24 B9 5C 19 B8 00 B5 E1 01 52 00 09 C0 AB
34032 CC 0F 34 B7 5C 3D 32 B7 5C 20 D4 C9 D0 4E 45 B0 14
34048 46 46 34 B6 5C CB 7F 20 03 ED B0 C9 14 BE 20 08 48
34064 23 13 08 7B B1 20 F5 C9 E7 15 3D CB 18 46 2B 19 B8
34080 00 CB 43 CE CD D2 10 BD 7E 19 C0 78 09 06 00 0D 08 32
34096 66 1B D0 6E 14 4E C0 60 0A AF C0 78 09 C5 FC 08 32
34112 0D 2A 4F 5C 11 14 00 0D 19 0D 7E 00 FE B0 3E 00 84
34128 C4 78 09 0D 7E 0A FE C0 20 05 CD FC 08 18 E1 DD 44
34144 5E 09 D8 56 04 D8 19 1B B0 7A B3 C8 CD 6F 0F C9 9B
34160 CD 1C 0B DD 7E 19 CD 7B 09 CD 9F 10 0D E5 E1 11 86
34176 52 00 19 01 01 00 CD 21 0A CD 8A 0E 3B 02 E7 11 FC
34192 BD 77 29 C3 2E 11 DD 66 1B DB 6E 16 01 01 00 03 51
34208 CB D1 1D 23 7B 20 FE 3F 28 F5 4F C3 78 09 C3 FC 40
34224 0B E5 C5 CD 21 0A 23 23 06 44 7E E6 3F DD 8E 29 04
34240 20 02 36 3F 23 23 23 10 F1 C1 E1 C9 60 0A C9 C5 68
34256 01 7F 00 ED 78 E6 40 C1 C8 E7 0E 8D 21 FA FF DD 50
34272 19 0D C8 1B 46 20 02 B7 0C DD 5E 0B 0C DD 96 96
34288 B5 00 19 DD 77 52 DD E1 13 DD 73 0B 0D 72 0C CB D3 3D
34304 44 CB DD 7E 19 C0 7B D9 CD B0 10 AF C0 78 09 C9 57
34320 CD 50 11 3B 14 F0 CB 7C 6E 28 05 C0 43 10 E7 0F 6F
34336 DD 7E 00 A7 CC 2E 11 B7 0F 1B 00 7E 0C 0F 0F FE 48
34352 B0 20 0B 0B C8 43 4E 28 02 CB F7 D0 B6 29 12 13 4E
34368 0D 7E 0D 12 13 DD 7E 0B 12 0D E5 E1 11 52 00 19 24
34384 CD 60 0A 0D 34 0D AF DD 77 0B 0D 77 0C C9 0D B5 4E
34400 E1 11 52 00 19 01 01 00 E5 CD 21 0A 11 0A 00 0D 35
34416 46 29 3E 2B 90 47 19 10 FD 36 3F 3F CD 60 0A C9 2B
34432 0D CB 29 FE C0 84 0B DB CB 29 BE C9 34 DB 5C 37 20
34448 D1 16 3A B6 5C 32 CF 5C 3E 05 32 D6 5C CB 1C 0B 7E
34464 DB E5 06 CB DD 36 52 00 DD 23 10 F8 DD E1 3A CF C4
34480 5C CD 78 D9 01 D1 0D 8D 6E 1A DB 66 1B 03 CD 21 60
34496 0A CD 70 12 23 78 2B FE 3F 28 F2 01 01 DD CD 21 6C
34512 0A AF CD 78 09 E5 21 21 12 CD 3F 12 3A CF 5C C6 89
34528 30 CD 5D 12 3E 3A CD 50 12 E1 E5 23 CD 3F 12 CD DA
34544 19 12 CD 19 12 AF CD 77 28 06 2D 7E FE 3F 20 24 80
34560 11 0B 00 19 1D F5 DD 7E 28 A7 28 D3 CD 19 12 E1 68
34576 21 28 12 CD 19 12 CD 3F 12 06 3F CD 5B 12 CD 19 86
34592 [illegible]
34608 12 D0 7E 28 A7 28 0B AF DD 77 28 3E 0D 18 05 3E 32
34624 06 DB 77 28 CD 50 12 18 BB F5 3E 0D CD 50 12 F1 2A
34640 C9 01 43 A1 54 41 4C 4F 47 20 23 A4 52 45 45 20 56
34656 52 45 43 3A 20 DD E5 C5 D7 28 1A C1 DD E1 C9 C5 81

34672 06 0A 7E CD 5D 12 23 10 F9 E5 D7 A8 0B E1 C1 C9 6B
34688 DD E5 37 10 00 DD E1 C9 C5 D8 E5 16 00 5B E5 D8 E7
34704 E5 E1 19 19 11 52 00 19 C8 35 12 E1 DD E1 C1 C9 81
34720 C5 06 AA E5 23 23 16 00 7E 86 3F 5F 8D E5 DD 19 70
34736 DD 19 DD 34 53 DD 7E 53 FE 00 20 03 DD 34 52 DD 69
34752 E1 23 23 23 10 B2 E1 C1 C9 3A 5B 5C B7 21 16 5C 2F
34768 5F 16 00 19 E5 CD 3B 0D 0B 0B CB 0A BE AF C8 7B 09 BC
34784 E8 81 73 23 72 C9 87 27 17 79 80 CB C5 E5 2A 4F C6
34800 5C 28 09 B3 D7 EB 16 2A 4F 5C 11 14 00 19 81 37 6A
34816 BD 52 C1 00 C5 85 88 22 51 5C 11 04 00 19 7B 13 E3
34832 19 5B 23 56 D5 F8 42 2B 16 FE 54 2B 12 FB 4D 20 3A
34848 19 D1 DD E1 D1 FD CB 7C 4E CA 8A 0F C3 FC 0B FD 95
34864 CB 7C 4E 20 05 3E 0D CD C9 16 C1 B1 C5 87 8D 19 F0
34880 AF 21 16 5C 5E 23 56 2B 22 5F 5C C1 E1 E5 C5 A7 1A
34896 ED 52 30 0D EB A7 ED 42 EB 2A 5F 5C 73 23 72 2A 3B
34912 5F 5C 23 23 3C FE 10 38 DB FD 36 26 00 E1 B1 FD 76
34928 CB 7C 8E C9 FD CB 7C E6 CD A8 13 2A 4F 5C 85 CD D7
34944 F7 13 CD AB 13 CD F7 13 D1 2A 4F 5C B7 ED 52 DD F2
34960 58 DA 5C 19 22 DA 5C 2A DA 5C 22 51 5C B7 E6 15 B3
34976 3B 04 2B F9 1B 08 2A E2 5C 22 51 5C D7 10 00 1B B6
34992 E6 FD CB 7C A6 2A 4F 5C E5 CD F7 13 C8 E5 13 CD F3
35008 F7 13 D1 2A 4F 5C 87 E8 52 ED 58 DA 5C 19 22 DA 3D
35024 5C CD E5 13 CD 10 10 C9 3A D8 5C 3C 28 0B 3E D7 C8
35040 01 16 2A 51 5C 22 DA 5C C9 3A D9 5C FE 4D 20 0C 85
35056 CD 3B 0D 4F CD 7B 09 DE 22 84 5C C9 FE 54 28 06 DD
35072 FE 42 2B 02 E7 00 CD 34 16 ED 53 D4 5C 85 D8 E1 77
35088 3D CD 04 FE C9 3A D8 5C 3C CD 34 0D 5C FE 4B 00 5C
35104 DD 2A 8A 5C C3 E4 0F 11 86 5C 21 0E 5C 06 0B 14 BF
35120 4E E0 71 12 E0 23 13 10 F6 C9 EB 24 51 5C 21 13 B6
35136 14 1B 4B DD CB 1B 46 28 02 E7 0D 88 5B 0D DB 56 14
35152 0C DD 6E 45 DD 66 46 37 ED 52 3B 13 13 8B 73 DB 54
35168 D2 7C 1B D0 E5 D0 19 D0 7E 52 DD E1 37 C9 D0 76
35184 CB 43 4E 28 04 4F C6 0D C9 11 00 00 D0 73 0B DD 1C
35200 72 0C DD 34 00 C0 40 00 4F C0 79 09 1B B0 FD CB B6
35216 02 9E E5 24 30 5C 5E 23 56 A7 21 7F 10 E0 52 20 D5
35232 2B D1 ED 7B 30 5C D1 01 BD 53 3D 5C E5 11 B2 14 14
35248 05 E9 3B 0A 2B 05 F0 36 00 07 EF E1 1B EB FE 00 48
35264 28 06 07 85 0F E1 1B E4 E1 C3 08 08 81 11 A2 14 B2
35280 05 E9 0B CB F0 CB 7C 66 2B 0C F6 0F C9 22 C7 5C 11
35296 E1 22 C9 5C E1 E5 22 CB 5C 24 C9 5C B5 2A C7 5C B6
35312 CD 1C 0B DD 7E 19 CD 7B 09 01 7F 00 ED 78 E6 40 AB
35328 40 28 02 E7 CB 3E DD ED 79 C9 9A 0A BD FB 1F 3D B2
35344 FB AF DD E5 E1 11 52 00 19 EB 21 14 16 01 26 00 26
35360 ED B0 F5 CD 70 15 CD D7 15 0E 01 F1 F5 47 0D 66 1C
35376 1B D8 6E 1A CD 21 0A 0C 79 FE 0B 20 F1 F1 3C F8 42
35392 2B 2B 05 CB F5 15 1B D6 DD E5 06 04 D8 6E 14 D8 32
35408 66 1B E5 23 B0 7E 0E 77 DD 23 23 10 F7 E1 01 01 76
35424 00 C0 60 04 C5 85 E5 81 13 36 3F 01 FF 01 80 80 BD
35440 E1 C1 03 71 CD 60 04 03 71 C8 60 0A E5 C5 11 9B 4B
35456 00 19 01 67 01 E5 D1 13 36 00 E8 B0 C1 E1 03 71 34
35472 23 36 00 2B CD 60 0A BD E1 4F CD 7B 09 C3 FC 03 40
35488 F0 E5 D0 E5 D0 B5 F0 B1 11 52 00 B8 19 C8 77 5B 6C
35504 4F FD 77 61 21 B8 B6 0B 04 BD 7E 01 FE A6 2B 23 B6
35520 FE 3F 28 0D DD 46 00 77 23 10 FC B0 23 B3 23 1B B7
35536 E5 00 23 0D 23 0D 28 22 F0 B5 D0 B1 11 54 0D BD 21
35552 19 1B 06 00 23 0D 23 0D 5E 01 16 00 21 08 3B 3D
35568 16 19 7E 00 77 01 E1 D1 1B BF 0E 4E 71 23 7C B5 AC
35584 20 F4 0D E1 FD B1 C9 21 08 06 16 00 1B C1 01 7F C3
35600 00 3E F4 E0 79 43 ED A3 CD 9A 0A BD 43 42 E0 78 09
35616 1F 3F F8 17 C9 C5 F5 3E 50 01 7F D0 ED 79 C0 94 C1
35632 04 ED 7B 1F 3B F8 F1 C1 C9 01 06 D7 D8 09 0A 04 68
35648 05 01 02 03 50 4E 0D 00 03 F5 01 FE 01 00 01 00 AA
35664 00 46 01 00 01 02 01 F7 16 48 0C 00 03 F5 01 F8 B6
35680 00 3F 00 3F 01 F7 12 4E 00 43 24 53 5C 2B 01 0B 85
35696 00 C5 D7 55 16 C1 C5 D5 C8 78 0C D1 C1 21 93 16 1A
35712 E0 B8 13 3A B9 5C F8 42 CD FE FF FE FF 05 00 19 11 C9 15
35728 16 73 23 72 23 11 94 16 73 23 72 81 C9 CD 3A 16 C1
35744 24 4F 5C 23 E8 4F 47 BD 52 EB 21 16 5C 3A B6 5C 0C
35760 07 4F 09 73 23 72 C3 C1 05 08 00 0B 00 54 AB 16 15
35776 94 16 08 00 21 A0 16 C3 5B 14 21 44 16 C3 5E 14 B1
35792 06 1F 18 02 06 3F DB 1F A0 37 C9 FE A5 3B 06 B6 B5
35808 A5 D7 10 0C C9 FE 7F 38 02 38 3F FE 0D 20 07 CD B6
35824 C9 16 38 0A 18 03 FE 20 D8 F5 CD 0D 0A F1 CD 8A 1F C8 D6
35840 6F 20 F7 F1 D3 5F D3 FE 3A C6 5C E6 0A D3 3F F6 CF
35856 04 03 3F C9 FF FF FF FF FF FF FF FF FF FF FF FF D3
35872 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF F0
35888 [illegible]
35904 7C 3E C3 33 00 FF FF FF FF FF FF FF FF FF FF FF F5
35920 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF F0
35936 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF F0
35952 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF F0
```


kozmetika Dahlia kozmetika

u selekciji najboljih . . .

loto na „spektrumu"

Ako vas neko pita koji je dobar sistem za loto nemojte mu odgovoriti da je to onaj koji donosi prvu nagradu. Nijedan sistem sam po sebi ne može doneti zgoditak. Sistem samo rasporeduje brojeva koje je čovek odredio. Od toga koji su brojevi određeni znači od čoveka zavisi koliko će biti pogodaka u sistemu. Zadatak sistema je samo u tome da pogođene brojeve, skupi u jednoj kombinaciji.

Ukoliko je sistem pun, svi pogođeni brojevi će se obavezno naći na jednom mestu. Ukoliko je skraćen onda će se bar u jednoj kombinaciji naći minimum pogođenih brojeva koji sistem garantuje a s nešto sreće i više od tog minimuma.

Odgovor je jasan, ali namaće novo pitanje.

Šta je pun, a šta skraćen sistem?

Vrtoglava progresija

Želimo da odigramo, na primer, osam brojeva. Neka su to brojevi od 1 do 8. Od osam brojeva moguće je napraviti osam kombinacija za loto (kombinatorika ih naziva kombinacijame bez ponavljanja sedmog razreda od osam elemenata). U našem slučaju pun sistem će imati sledeće kombinacije:

1 2 3 4 5 6 7
2. 1 2 3 4 5 6 8
3. 1 2 3 4 5 7 8
4 1 2 3 4 6 7 8
5. 1 2 3 5 6 7 8
6. 1 2 4 5 6 7 8
7 1 3 4 5 6 7 8
8. 2 3 4 5 6 7 8

Izdvajamo sada dve kombinacije, petu i osmu

5. 1 2 3 5 6 7 8
8 2 3 4 5 6 7 8

Te dve kombinacije čine skraćeni sistem koji garantuje u najnepovoljnijem slučaju, dva puta po šest pogodaka od sedam pogođenih brojeva. Drugim rečima ako bi bila izvučena ma koja od ostalih šest kombinacija mi bismo imali dva puta ovu manju nagradu, a prve nagrada bi nam pripala ukoliko bi bila izvučena jedna od dve kombinacije skraćenog sistema.

Skraćeni sistem nam znači, omogućuje da uz višestruko smanjenu uplatu imamo neki garantovani minimalni dobitak — i nadu da će se uklopiti sedmica' Naravno, pod uslovom da sa medu brojevima koje smo odigrali nađe onih sedam koje će Suzana Mančić izvući sledećeg utorka.

I tako, sve ovo izgleda lako i jednostavno kad je u pitanju osam brojeva i isto toliko kombinacija. Ali broj kombinacija u lotu vrtoglavo raste. Pun sistem od 10 brojeva ima 120 kombinacija, od 20 brojeva 77.620 od 30 brojeva 2 035 800, a od 39 brojeva svih 15 380 937 kombinacija. Kako od tolikog broja kombinacija izdvojiti onih nekoliko koje će nam pružiti bar kakvu-takvu garanciju?

Većina matematički obrazovanih ljudi, s dobrim poznavanjem kombinatorika slagnula bi ramenima kad biamo od njih zatražili odgovor na ovo pitanje. Jednostavno nisu sa bavili tim problemom a za potpun odgovor verovatno ne bi bilo dovoljno stranica u ovom broju Računara. Zato nije nikakvo čudo što se u Jugoslaviji bezmalo na prstima jedna ruka mogu izbrojati ljudi koji prave skraćene sistema za loto. Po abecednom redu to su: Bećirović (službenik) Jokić (arhitekt), Joksimović (Inženjar) Kokotović (akonomist) Šćepanović (inžanjar) možda još dvojica, trojica kojima se skromno pridružuje i dolepotpisani hakar Ž. V.

Ukoliko nameravate da upiovite u vode tog područja kombinatorika očigledno morate biti spremni za duboko ronjenje. Ali, ako su vaše ambicija samo igračke, a imate spektrum avo rešenjal Ukucajte bajzik i mašinac sa ovih stranica, startujte program — i moći ćete da napravite hiljade različitih punih i skraćenih, uslovnih i bezuslovnih sistema.

Temo-amo po meniju

Program „Sistemi za loto" ima sledeće mogućnosti:

a) Generisanje punih sistema.

b) izrada bezuslovnih skraćenih sistema a garancijama 4, 5 i 6 od 7

c) izrada uslovnih skraćenih sistema a garancijame 4, 5, 6 i 7 od 7

Sve sisteme program prikazuje u kombinacijame po 7 brojeva.

Broj kombinacija u jednom sistemu na može biti veći od 4 000, što je uslovljeno raspoloživom memorijom računara, ali broj brojeva nije ograničen ničim osim pravilima loto.

Svi podaci se upisuju samo pomoću dve dirke: SPACE i ENTER. SPACE služi za biranja podataka, a ENTER za unošenje.

Kad program proradi, na ekranu ćete ugledati meni sa sledećim opcijama:

1. Broj brojeva

Možete praviti sistem od 8 do 39 brojeva. Ova opcija je obavezna. Ako ne odredita koliko će sistem imati brojeva program će prihvatiti izbor drugih opcija, osim opcija 2 i 15.

2 Garancija

Takođe obavezna opcija. Najniža moguća garanoija je 4 od 7, najviša 7 od 7. Za neke vrste sistema dovoljni su samo podaci iz prve dve opcije. Korišćenje ostalih opcija nije obavezno.

3. Zamene brojeva

Ova opcija omogućuja određivanje brojeva pra izrada sistema. Ako, na primer pravite sistem od 12 brojeva možete odabrati bilo kojih 12 brojeva od 1 do 39. Ukoliko brojave na zamanite, računar će prikazati sistem brojevima od 1 do 12. Zamena brojeva bitno utiče na broj kombinacija ukoliko sa primeni jedna od ovih opcija: 6, 7, 8, 13 ili 14.

4 Fiksiranje

Možete tikairati 1, 2 ili 3 broja. Time ćete postići da program iz punog sistema izbaci kombinacije koje na sadrže sve fikairane brojeve.

5. Favorizovanje

Program omogućuje favorizovanje 2 do 5 brojeva, a iz punog sistema će izdvojiti u skraćeni samo kombinacije koje sadrže najmanje jedan favorizovan broj. U jednom sistemu možete istovremeno primeniti fiksiranje i favorizovanje. Vodite samo računa da vam isti broj ne bude i fikairan i favorizovan jer je to besmisleno.

6. Razmak

Ovom opcijom sa određuje minimalni dozvoljani razmak izmedu brojeva u sistemu. Mogući minimalni razmaci su 1, 2 i 3. Na smete u istom sistemu upotrebiti ovu i opciju broj 7.

7. Blizanci

Ako niste primenili opciju 6, možete odrediti maksimalan broj takozvanih blizanaca (perove susednih brojeva na primer 3—4). U jednoj kombinacliji mogu sa dozvoliti najviše 1, 2 ili 3 blizanca.

8. Par/neper

Možete odrediti maksimalan broj parnih i neparnih brojeva u jednoj kombinaciji. Oba maksimuma ne moraju biti iste ali njihov zbir na sma biti manji od 7.

9. Prvi broj

Ovom opcijom sa ograničava opseg prvog (najmanjeg) broja u kombinacliji. Možete odrediti, na primer, da tej broj bude u opsegu od 1 do 10.

10. Poslednji broj

Ograničavanja opsega poslednjag broja u kombinacliji.

11 Dve grupe

Ovom opcijom brojevi u sistemu se dele u dve grupe. Sami određujete koliko će brojeva biti u kojoj grupi, a program rasporeduje brojeva u kombinacliji po obrascu 3+4 (3 broja iz prve i 4 iz druga grupe). Brojeve na smete istovremeno deliti u dve i tri grupe.

12. Tri grupe

Ako brojeve podelite u tri grupe njihov raspored u kombinacliji će biti po obrascu 2+2+3. Obratite pažnju na ograničavanje u vezi sa opcijom 11.

13. Tri kolone

I još jedna podela brojeva — na kolone onako kako su odštampani na tiketu. Vi određujete minimalan broj brojeva u kombinaciji iz avaka kolone. Uz ovaj uslov možete koristiti i uslove iz opcije 11 ili 12.

14 Zbir brojeva

Ovom opcijom određuje se minimalni maksimalni zbir brojeva u kombinacliji. Program će iz punog sistema eliminisati sve kombinacije čiji zbir brojeva izlazi iz dozvoljenog opsega.

15 Izlaz

Biranjem ove opcije izlazite iz menija i pokrećete mašinaku rutinu koja će napraviti skraćeni sistem prema vašim zahtevima.

Eksperт za loto sistеme i strastveni programer na „spektrumu" Žarko Vukosavljević pripremio je pre nekoliko meseci za čitaoce „Računara" program koji bi trebalo da im olakša dilеme „kako-popuniti-loto-listić-za-utorak-uveče", ali je redakcija, verovatno zbog preterane skloпosti ka programerskim temama, ovaj projekat odlagala nekoliko meseci. Od ovog programa, razume se, ne treba očekivati čudo tipa „kako-dobiti-pet-milijardi-na-lotou" — njegova osnovna namena je da pomogne igraćima da, uz povećana šansa na dobitak, racionalizuju svoje ulaganja. Iz tehničkih razloga program smo podelili na dva dela — u ovom broju objavljujemo mašinska rutina, a u sledećem program na bejziku.

```
63501  21 B9 88 11 BA BB D1 60 6B 36 00 ED B0 34 EB FD  75
63517  FE 01 CC BA FB FE 02 CC 91 EB 3A EA FD FE 01 CC  9A
63533  9B FB FE 02 CC A4 FB FE 03 CC B1 FB 21 19 F6 06  AB
63549  27 7E FE FF 28 06 23 10 FB 00 00 00 3E 02 CD 01  09
63565  16 21 EC FB CB 71 FB 21 01 00 22 B2 FF 06 07 3A  93
63581  7B FF 21 BB FF 77 3D 28 10 FB 06 07 3E 01 21 BF  6B
63597  FF 77 3C 23 10 FB BA AB FF FE 07 28 06 3A 7B FF  A5
63613  32 C5 FF 21 BF FF 11 AB FF 06 07 7E 12 23 13 10  73
63629  FA C3 2C F9 3A 0B 5C FB 53 CA DC FC 3A FF 5A 3C  72
63645  32 FF 5A 11 B1 FF 21 BB FF 1A BE 28 05 3C 12 C3  38
63661  2C F9 11 B0 FF 21 BA FF 1A BE 28 0B 3C 12 BE CA  92
63677  2C F9 1B 6B 11 AF FF 21 B9 FF 1A BE 28 DB 3C 12  93
63693  BE CA 2C F9 1B 53 11 AE FF 21 B8 FF 1A BE 28 0B  B6
63709  3C 12 BE CA 2C F9 1B 3E 11 AB FF 21 B7 FF 1A BE  BD
63725  28 0B 3C 12 BE CA 2C F9 1B 29 11 AC FF 21 B6 FF  FE
63741  1A BE 28 0B 3C 12 BE CA 2C F9 1B 14 11 AB FF 21  0B
63757  B5 FF 1A BE CA 0C FC 3C 12 BE CA 2C F9 3C 13 12  BA
63773  3C 13 12 3C 13 12 3C 13 12 3C 13 12 3C 13 12 21  06
63789  AB FF 11 A1 FF 06 07 7E CD 7B FB 12 23 13 10 F7  7B
63805  21 D4 FF AF B6 21 07 3A 41 FF BE B2 91 F8 21 0B  7B
63821  FF AF B6 28 07 3A A7 FF BE BA 91 F8 21 B4 FF AF  17
63837  B6 28 22 7E 57 0E 00 21 A1 FF 06 07 CB BD 21 CC  B5
63853  FF DD 7E 00 BE CC BB FB BB 23 1B 20 F4 23 10 EC  B7
63869  79 BA C2 91 FB 21 BE FF AF B6 28 1B 7E 57 21 A1  9D
63885  FF 06 07 4A DD 21 CF FF BD 7E 00 BE 28 0B DD 23  6E
63901  0B 20 F5 23 10 ED C3 91 FB 21 D4 FF AF B6 28 10  1F
63917  7E 47 21 A7 FF 06 06 7E 2B 96 B9 0A 91 FB 10 F7  02
63933  3A B6 FF B7 28 2A 0E 00 21 A1 FF 06 7E 23 3C 00  DD
63949  BE CC BB FB 10 F7 3A D6 FF B9 DA 91 FB 11 A0 FF  EF
63965  06 05 13 B5 E1 7E 23 23 3C 3C BE CA 91 FB 10 F2  23
63981  3A D7 FF B7 28 21 21 A1 FF 0E 00 06 07 CB 46 CC  C9
63997  BB FB 23 10 FB 3A BB FF B9 BA 91 FB 3E 07 91 4F  00
64013  3A D9 FF B9 D4 91 FB 21 BB FD AF B6 28 13 00 2A  DB
64029  F2 FD 23 0D 7B 00 BE B2 91 FB 00 7E 01 BE 0A 91  0B
64045  FB 21 EA FB AF B6 28 26 0D 2A F2 FB 23 00 7E 00  27
64061  BE B2 91 FB 0D 7E 01 BE 0A 91 FB B0 24 F4 F0 23  31
64077  B0 7E 00 BE D2 91 FB 0D 7E 01 BE 0A 91 FB 3A E0  1B
64093  F0 B7 28 6C AF 32 B5 FF 0E 07 11 A1 FF 21 C5 FB  A4
64109  06 00 1A BE CC BB FB 23 10 FB 13 00 20 EF 21 E0  D7
```

```
64125  FF 3A D[illegible]  A1 FF 21 [illegible]
64141  FF 21 [illegible]  06 0D 14 BE CC BB [illegible]
64157  20 EF 21 EE FD 3A D5 FF BE D4 [illegible]
64173  0E 07 11 A1 FF 21 DF FB 06 0D 1A BE [illegible]
64189  10 F8 13 0D 20 EF 21 EF FD 3A D8 FF BE [illegible]
64205  3A E0 FD B7 28 17 21 41 FF 0E 00 7F AF 4E 81 [illegible]
64221  FB 21 F0 FD BE DA 91 FB 23 81 D4 91 FB 21 [illegible]
64237  0E 07 4F 32 49 FF 11 41 FF 06 0D 1A BE 28 [illegible]
64253  10 F9 23 0B 20 F0 AF B6 20 E6 2C BC FF 1B [illegible]
64269  3A 4B FF 57 3A A9 FF 3C B4 28 06 32 49 FF D[illegible]
64285  BE D1 C3 91 FB 21 AB FF 11 BF FF 06 07 7E 1[illegible]
64301  13 10 F4 21 A1 FF ED 5B BC FF 06 07 7E 12 [illegible]
64317  10 FA CD 51 FB 7B FE 0F C2 8D 18 79 FE 4D C[illegible]
64333  FB C3 0C FC 3E 16 CD 1D 00 3E 00 CD 1D 00 3E [illegible]
64349  C8 10 00 ED 4B B2 FF CB 18 1A ED 4B B2 FF 03 [illegible]
64365  43 B2 FF C9 7E FE 40 C8 CD 1D 00 23 18 F6 BD [illegible]
64381  7B FF BB 23 3D 20 FB DD 7E 00 C9 0C C9 21 A0 [illegible]
64397  22 F2 FB C9 21 AE FF 22 F2 FB C9 21 4C FF 22 F[illegible]
64413  FD 23 23 22 F4 FD C9 21 4C FF 22 F2 FD 23 23 [illegible]
64429  22 F4 FD C9 21 A0 FF 22 F2 FD 23 23 22 F4 FD C[illegible]
64445  3A D5 FF 3C 32 D5 FF C9 01 04 07 04 0D 10 1[illegible]
64461  19 1C 1F 22 25 02 05 08 0B 0E 11 14 17 1A 1D 2[illegible]
64477  23 26 03 06 09 0C 0F 12 15 18 1B 1E 21 24 27 [illegible]
64493  0D 00 A2 52 AF 4A 20 48 AF 4D 42 49 4E 41 43 [illegible]
64509  A4 41 3A 30 40 21 74 00 01 5B 01 C3 00 00 C9 [illegible]
64525  32 D5 FF 21 7B FF 46 23 10 FC C9 3E 14 BE 28 0[illegible]
64541  14 B7 2B 06 1B F5 23 10 EF C9 3E FF 32 D5 FF [illegible]
64557  00 3A AA FF B7 20 17 3C 32 AA FF 21 CD BB 2A [illegible]
64573  FF 21 00 00 22 C6 FF 2A B2 FF 2B 22 C4 FF 3E [illegible]
64589  CB 01 16 16 FF 1E 10 2A CB FF CD BC FC 06 07 [illegible]
64605  FE 0A 30 07 F5 3E 20 CD 10 00 F1 C5 06 0D 4F [illegible]
64621  1B 1A C1 3E 20 C5 10 00 23 10 E4 E0 4B C4 FF [illegible]
64637  E0 43 C4 FF 78 81 2B 03 10 20 CF 22 CB FF C[illegible]
64653  CC BE FF C0 1B 14 3E 2E C5 10 00 3E 20 CD 10 0[illegible]
64701  C9 79 FE 64 30 FE 04 30 05 3E 20 CD 10 00 3E [illegible]
64717  CB 10 00 C9 00 00 00 00 00 00 00 00 00 00 00 [illegible]
64733  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [illegible]
```

Dok se sistem pravi indikator rada u donjem desnom uglu ekrana obaveštava vas promenom boja i treptanjem, da je sve u redu. U gornjem delu ekrana računer broji kombinacija koje unosi u skraćeni sistem. Izradu sistema možete prekinuti u svakom trenutku pritiskom na S(top), a zatim sa vratiti u meni radi dopuna ili izmene podataka, ili na start programe. Zvučni indikator obaveštiće vas o kraju izrade sistema. Tada ćete imati prilku da donosite sledeće odluke: D (dalje, opcija za razgledenje sistema), P (ponovni print, broj razgledanja nije ograničen), C (copy, kopiranje kombinacija a ekrana printerom), M (povratak u meni), R (restart: pogledajte opciju 16).

16 Restart

Vraća program na početak i briše sve ranije unesena podatka.

Nevolje u početku

Delavaće se, naročito u početku da vas program izvesti kako se sistem ne može napraviti. Uzrok tome će biti, najčešće, rdavo kombinovanje uslova. Na primer:

Tražili ste garanciju 4 od 7 za sistem od 9 brojeva. To je malo brojeva za postavljeni zahtev. Morate povećati broj brojeva ili tražiti veću garanciju.

Odredili ste minimalni razmak 2 za sistem od 19 brojeva. Ako niste obavili zamenu, ovakav sistem mora imati najmanje 21 broj i to za garanciju 7 od 7.

Zahtevali ste najmanje 3 broje iz druge koiona (opcija 13) a u sistemu imate samo dva takva broja.

63/loto na „spektrumu"

Ovo su samo tri od bezbroj mogućnosti da pogrešite. Lako su uočljiva jer se odnose na lošu kompoziciju samo dva uslova. Šta se tek može desiti kad se pogrešno ukrste uslovi iz nekoliko opcija, teško je pradvideti. Zato je najbolja da se na program privikavate pravaći sistеme sa malo brojeva, do 15. Rezultate ćete dobijati brzo, pa ćete lako uočiti uticaj pojedinih uslova na broj kombinacija. Tak posle toga sa možete upustiti u avanturu što je pravljenja sistema od 39 brojeva sa 4 000 kombinacija.

Prethodno pripremita jastuk da odspavate dok spektrum radi. Mašinac jeste brz ali ovde ima mnogo posla. Da bi ispitao sličnost dve kombinacija mora da obavi 49 poređenja (7 puta 7). Ali pravi sistem od 39 brojeva, samo za ispitivanje sličnosti obavićе sledeći broj poređenje broj kombinacija punog sistema (15 380 937) puta broj kombinacija skraćenog sistema (recimo 1 000) puta 49. A gde su ostali zadaci? Na primer prauzimanje podataka od bejzika, ispitivanje u vezi sa svim postavljenim uslovima brojanje i memorisanje odabranih kombinacija, promena atributa u indikatoru rada testiranje tastatura, ispitivanje ostatka slobodne memorije.

Posebne mašinska rutina, po završetku rada prve printuja kombinacije skraćenog sistema. To bi ukratko, bio opis najvažnijih poslova koja obavlja mašinac. A što se tiče bejzika, verovatno ste već uočili njegov zadatak. A što se tiče bejzika, verovatno ata već uočili njegov zadatak da prihvati podatke o sistemu i da ih prosledi na izvršavanje brže od seba, mašinci, kao i da podnese eventualna izveštaja na osnovu kodova koje je ostavio mašinski program.

Evo i nekoliko testova za vreme:

12 brojeva, garancija 6 od 7 [illegible] brojeva do 4, nepermih brojeva do 4 [illegible] na dve grupe po šest brojeva. 5 [illegible]

15 brojeva, garancije 5 od 7 [illegible] jedan broj. 24 sekunda.

18 brojeva garancija 4 od 7 [illegible] favorita: 1 minut i 34 sekunde.

20 brojeva garancija 4 od 7 bez uslova: 8 minuta i 30 sekundi.

22 broja garancija 4 od 7 bez uslova: 25 minuta i 45 sekundi.

28 brojeva garancija 5 od 7 bez uslova: celu noć a možda i parče jutra.

Ako napravite ova sisteme možda nećete dobiti sedmicu ali ćeta bar saznati de [illegible] ULA čip dober to jest da li možda da vaš dugotrajni rad bez posledica zbog pregrevanja.

Program Sistemi za loto" ne daje [illegible] najmanji mogući broj kombinacija. Za uzvrat kombinovanjem više uslova možete postići izuzetno velika skraćenja.

Nake sisteme, na primer one s garancijom tipa 6 od 6 5 od 5 itd. kao i [illegible] sa pišu na sistemskim listićima (po [illegible] više brojeva u kombinaciji) ovaj [illegible] na da napravi. No to ne znači da već nisu napisani programi koji izrađuju [illegible] sisteme. Zagrižienim loto-hakerima [illegible] prako redakcije Računara stoji na raspolaganju za savete i uputstva (podrazumeva se bez naknade).

Žarko Vukosavljević

U domaćoj radinosti

Proširenje memorije za „spectrum"

korak ka plusu

U ostroj konkurenciji na tržištu računara proizvođači se na svaki način trude da im proizvod bude što jeftiniji, često zaboravljajući na kvalitet i standarde. Setimo se kompjutera sa samo jednim kilobajtom RAM-a, raznih membranskih tastatura ili „ametrade" sa svojim troičnim disk-drajvom. Apsolutni šampion u škrtarenju je, bez sumnje, ser Klajv Sinkler. Nestandardni priključci i „revolucionarne" tastature su već ušle u poslovice, ali se manje zna o tome da, recimo u „spektrum" ugrađuje nelepravne memorije. Memorijske čipove od 32 K na proizvodi niko u svetu (a ni kod nas), već su to memorije od 64 K kojima je jedna polovina nelepravna. Tako je ovaj engleski inovator dobio plemićku titulu, a „spektrum" ostao sa 48K RAM-a. Odatle ideja da mu se doda još 16K i tako upotpuni adresni prostor procesora Z80 na 64 K RAM-a.

Osmobitni procesor može da adresire maksimalno 64 K memorije a mi želimo da „spektrum ima 64K RAM-a, plus 16K ROM-a. Znači neizbežno je preklapanje dodatnog RAM-a i ROM-a što automatski znači da dodatna memorija neće biti pristupačna iz bejzika. Preme tome eko još neste saviđeli mašinski jezik procesora Z80 od ove gradnje nećete imati nikakve koristi, ali ako ga znete, one će vam itekako dobro doći.

Proširenje memorije je zamišljeno kao kartica koje sa priključuje na spektrumov konektor. Električne šeme je dete na slici. Upotrebljena su dve statičke CMOS memorije tipa 6264 kapaciteta po 8K, koje su nešto skuplje od dinamičkih ali im je upotreba mnogo jednostavnije zbog lakšeg vezivanje u kolo (nisu potrebni selektori tipa 74157) i nema potrebe za osvežavanjem. Porad memorijskih čipove potrebne su još tri TTL integrisana kola.

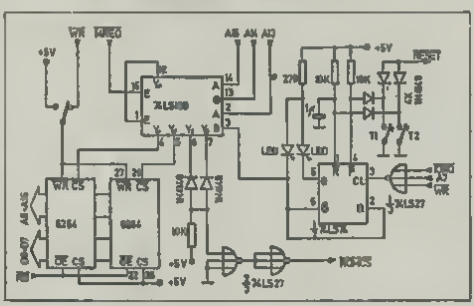
Dvostruki dekoder 74LS139 je upotrebljen za uključivinje jednog od dve SRAM-e ili ROM-a i to na sledeći način. Ulazi A i B dekodere, uz uslov da je kontrolni ulaz E na logičkoj nuli odrediju koji će od četiri izlaza biti takođe na nivou logičke nule. Tebele istinitosti za ovo kolo to obješnjave

E	B	A	Y0	Y1	Y2	Y3
1	x	x	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

Oznake upotrebljene u tebeli znače

0 — nivo logičke nule
1 — nivo logičke jedinice
x — nivo logičke nule ili jedinice, nebitno je

Kad procesor proziva naku od adrese koje sa nalazi u donjih 16K njegove memorijeke mape, izlazi MREQ, A18 i A14 su na nuli, što dovodi jedan od



izlaza dekodere (Y0 — Y3) na nivo logičke nule. Koji će od njih to biti zavisi od stanje na ulazima A i B (nožice 2 i 3). Ulaz B je vezan na izlaz Q̄ flip-flopa 74LS74. Kade je ovaj na nivou nule, uključen je SRAM izvod sa konaktore A13, vezan na ulaz A čekodare (nožica 2), odrediuje će li će biti uključeno donjih ili gornjih 8K SRAM-a. ROM će biti pozvan kad ulaz B doda na nivo jedinice. Dlodno i kolo registruje eko je naki od izlaza Y2 ili Y3 ne null i to sa preko dvostrukog invertora sa 74LS27 prenosi ne kontroni ulaz ROM-a, ROMCS na konaktoru. Priilkom uključivanje kompjutare flip-flop sa resetuje, izlaz Q̄ sa postavlja na nivo jedinice i uključuje se spektrumov ROM. Prelazak na dodetni RAM sa može ostvariti softverski i hardverski. Mešinaka instrukcije OUT (127), A bez obzira ne to šta se nalazi u akumuletoru, vrši prebacivanje sa ROM-e na RAM i obrnuto, već zavisno od toga šte je uključeno u trenutku devenje naredbe.

Herdvarsko uključivenje RAM-a izvodi se pritiskom ne taster T1. Time sa flip-flop setuje (Q̄ dolazi ne nuiu i uključuje RAM) i istovremeno sa resetuje procesor. Sede, zneči počinje de sa izvršave program od nulte adrese u RAM-u. Nerevno, potrebno ja pre pritieka na taster upisati naki program u tu memoriju. Hardversko uključivanje ROM-a nije moguće e de pri tome na dođe do resetovanje računera. Pritiekom ne taster T2 uključujemo ROM i istovremeno resetujemo kompjutar (što će izbrisati sadržaj kompletna memorije ceim 16K SRAM-a na proširenju) pa zbog toga treba tastera T1 i T2 fizički odvojiti da ne bi došlo do slučajnog pritiska na destruktivni T2.

Led dioda nisu neophodna za rad uređeje ali su tu zbog indikacije uključenosti RAM-a odnceno ROM-a.

Ostaje još da objasnimo prisustvo preklopnika PR koji očigledno služi da (kad sa prebaci na položaj +5V) spreči upisivanje u RAM – praktično pretvare RAM u ROM! Tej preklopnik je potreban eko želite da eksperimentišete sa „spektrumovim" ROM-om. Sigurno nećeta odoleti de njegov sadržaj upišete u RAM i ispravite bagove koje su cetavili Sinklerovi programeri, de popunite poslednji kilobajt koji su cetavili nelakorišćen. Upisivanje ROM-e u RAM možeta de izvedete, recimo, ovakvim programom

```
        DI
        LD HL, 16384
LOOP    DEC HL
        LD A,(HL)
        OUT (127), A
        LD (HL), A
        OUT (127), A
        LD A,H
        OR L
        JR NZ, LOOP
        EI
        RET
```

Sada je sadržaj RAM-a ekvivalentan sadržaju ROM-a i sledeća OUT 127 x bajzik instrukcije bi isključile ROM i uključila RAM, odakle bi trebalo će se nastavi rad „spektrumovog operetivnog eletema. Ali to sa na dešava iz nakog (samo Sinklerovim programerima poznatog razloga). ROM negde upisuje nešto na adrese u okviru prvih 16K što, narevno, njemu na može će promeni sadržaj, ali ako je sade tu RAM dešavaju sa čudna stvari. To je najjednostavnije izbeći tako što ćete posle prebacivanje sadržaje ROM-a u RAM prekiopnik PR prebaciti iz položaje WR u položaj +5V i tek onda četi OUT 127,x instrukciju. Preklopnik možete do koristite i u drugim slučajevima kad želite da izbegnete mogućnost da (recimo, usled greške u nekoj vašoj rutini) dođe do brisanje programe u SRAM-u.

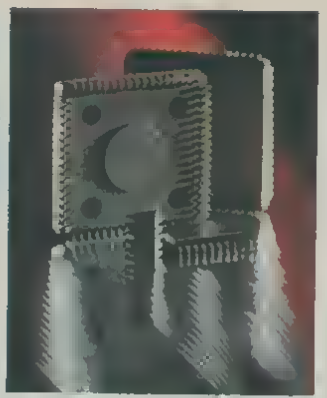
Ovom gradnjom nećete dobiti samo puko proširenje memorije već, pre svega, jednu korisnu sletku koje će vam znatno olakšati razvoj mašinskih programa. Ako u dodatni RAM spremite recimo GENS i SORSFAJL valeg programa, više neće biti potrebe de ih zbog toga što se računar blokiraio ili resetovao, ponovo učitavate se trake. Dovoljno će biti da pritisnete taster T1, u deliću sekunde prebacite program iz SRAM-a u RAM, i odmah se posvetite tražanju greške.

Pošto je ovo memorijeko proširenje namenjeno, uglavnom, iskusnijim korisnicima „spektruma", na dajemo nikakav softver za podršku. Kartica je višestruko primenljiva i verujemo da će svako prilagoditi ovaj uređaj i sfotver se njega svojim potrebama. To na važi za visnike interfejsa 1, jer ovaj dodatak nije kompatibilan s njim.

Miroslav Milošević

Šta ima novo

u svetu komponenata



Iako verovatno mnogim čitaocima UV EPROMI izgledaju kao (skoro) idealne memorijske komponenta, pa sa sa pravom (i) pitaju kako i zbog čega uspevaju da sa održe bipolarne jednokratno programabilne PROM memorije (jednom programiraj, i gotovo), verovatno su zaboravili da se UV EPROMI prave do mekalmalnih brzine pristupa do oko 150 ns, dok sa bipolarni PROM-ovi normalno nude sa vremenom pristupa reda 35 ns (firme MMI, harris, signetice i druge) Sada sa nude i CMOS PROM-ovi, iste brzine, sli sa manjom potrošnjom i širim temperaturnim opsegom (firme Cypress, Idt, AMD i dr)

Još sa nije osušile štamperska boja od prethodnog broja „Računare , u kojime smo pomenuli V20/V30/V40/V50 familiju 8- i 16- bitnih mikroprocesora, e sada nalazimo Informacije da Nippon Electric Company (NEC) nudi još složenije modele V60/V70 To su 32-bitne mašine realizovane CMOS tehnikom širine linija 1,5 mikrometara, i sadrže oko 375 000 trenzistora na čipu Svojim velikim sdresnim prostorom (4 Gbajta virtuelne mamorije) i zaštitnim mehanizmom veoma liče po erhitekturi ne INTEL-ove modele iAPX286 i iAPX386, pogotovu što su u pripremi i pripadajuća kola kao kiok drajver) generator, SUS kontroler, DMA kontroler, procesor za metematiku pomičnog zareza itd., kakve već postoje kod iNTEL-ove familije

Flopi diskove prečnika 2,5 inča nudi Hitachi Maxell, sa kapacitetom oko pola megabajta namenjen upotrebi u prenosnim tekat procesorime Metične firme (Hitachi) je proizvele disk drajv potreban za ovu novu vratu „flopiča – Semi diskovi su dimenzije 6X6 cm, a gustine zapisa je 200 tragova po inču (oko 50% guše od postojećih 3,5 inčnih diskova kapacitete 1 Mbajt)

Naravno ovde se radi o poznatoj firmi Surr-Brown koje nudi fantastične A/D i D/A konvertera uzmimo kao primer nove D/A pretvarače namenjene za upotrebu u audio uređajima (generisanje zvuka sintetičkim putem) sa oznakom PCM54 i PCM55. Ovi D/AC sa napajaju sa +/-12 odn +/-5 V, sadrže internu referencu, laserski trimovanu (podešenu) otpornu mrežu, brze strujne prekidače i izlazne pojačavače Uz garantovanu 15-bitnu monotonost koda, ove modele karakteriše fenomenalno niska tipična diferencijalne greške od 0,001%, dinamički opseg reda 96 dB, amirivanje naponskog izlaza za manje od 3 mikrosec, amirivanje strujnog izlaza u okviru +/-0,008% pune skale za samo 350 ns, disipacija ispod 200 mW, s ukupne harmoničke izobličenje su garantovano ispod 0,0025% pri punom ulazu, odnosno ispod 0,02% pri nivou od — 20 dB I sve to za oko 15 dolera, zavisno od tipe i pakovanje, impresivno, neme štal

Proizvođači elektronskih LSI komponenata nude širok izbor integrisanih kola eluže kao časovnik/kalendar (termin Rsal Time Clock), napr Motorole i Hitachi nude MC146818 (HD146818 respektivno), Netional Semiconounducter nudi MM58167/-MM58174/MM58274 Oki nudi MSM5832 itd. Obzirom da su predvidena za rad i po nestanku električne energije (Battery BackUp), po pravilu su izrađena CMOS tehnologijom sa potrošnjom rede mikroampera sadržava sve potrabne elementa osim kverc kristale i trimer-kondenzatora za fino podešavenje frakvencije Sade Epson (slično i Seiko) nudi IC sa oznakom RTC-58321 koje već sadrži kristal (32 kHz) podešeno je u fabrici na tacnu učestanost, sa garantovanim odstupanjem od 10 pom (deset delova u milion) Ove IC nieu skupe pe denes skoro de i neme personalnog računare koji neme ugrađen neki RTC

CODEC (COder/DECoder) je element, kao što znamo, koji analogni signal pretvara-kodira u digitalni niz, u ciju pranosa putem digitalnih veza, s zatim ponovo rastorira-dekodira niz u analogni signal Ovaj način pranosa ime višestruke prednosti veća imunost na smetnje, digitalni signal se leko obrađuje računarime, pogodnim izborom načine kodiranje može se signal šifrireti (sigurnosni aspekt), pranos svetlovodnim vlaknime itd CODEC tipa S2542 firme Slitronics koristi eksponencijalnu promenljivu delta modulaciju (algoritam) napajanje od 5 — 6V uz potrošnju od nekoliko desetine miliampera, e izrađeno je kombinacijom enalogne i digitalne (I^2L) tahnologije Brzine uzimenje uzoraka ide od 150 Kbps (tipično potrebno je oko 40 Kbps za telefoniju, e oko 100 Kbps za radio-difuziju)

Prame tvrdnji firme NEC Electronics njime pripada čast da eu prvi tržištu ponudili potpuno statičku RAM kapaciteta 256 Kbita Integrisano kolo sa oznakom uPD43256 ime 32Kx8 organizaciju, napaja se sa 5 V, ali sa čuvanje podataka garentuje i pri samo 2 V, e može sa dobiti u tri brzinske verzije 100, 120 i 150 ns Bazično čip koristi NMOS memorijske ćelije u metrici, e CMOS periferinu i pomoćnu logiku uz cenu rade 100 US dolera

Američka firme Advenced Micro Devices (AmD) je krajem 1985. obečale da će svake nedelje tržištu ponuditi novo, traženo LSI integrisano kolo (i) i to ne nešto jednostavno, nego komplekane kola kao što je npr Am8151 grafički čip za kolor paletu Ove LSI IC se koristi u vrhunskim grafičkim terminalime rezoluoije 1280 x 1024 (preko 2 500 000 tačaka), radi sa video signalem rede 200 MHz, ime izbor od 256 boja iz palete od oko 16,8 miliona boje, i slično Za sada AMD poštano „drži reč

$E^2 = MC$

Možde se Ajnštajn ne bi složio sa ovom formulom, ali Motorole dokazuje da je E^2PROM previ programski STORAGE za MC mikrokontrolere, leko eu neke varijante MC68HC11 serije sadržavale porad EPROM-a i RAM-a, i 512 do 2K EEPROM memorije tek pojave kontrolera MC68HC805C4 sa 4K bajta E^2PROM-a, u DVA nezavisne segmenta nudi mogućnost da softver „popravije sam sebe Naime dok se program izvršave iz segmenta A (EEPROMe), simultano sa može menjeti programeki kôd u segmentu B, pod programskom kontrolomi Uz ostale široka sposobnosti ovog MCU (RAM 24 I/0 linije, tajmer UART SPI itd), Motorola predvida sjajnu budućnost ovom čipu

Popularne statička RAM tipe 6116 (Hitachi) se može dobiti u raznim verzijama tako firma interesantnog imena Vitalic (ovo znači Very InTELligent Integrated Circuits!) nudi brzu verziju (vreme pristupa oko 55 ns), kao i štedljivu verziju (stend bej potrošnje oko 2 mikroampera) Oznaka modela V61C16, sa cenom reda 10 US $

Ako ste do ovog CPU i ostalih IC teško došli (s l u drugim slučajevima), preporučujemo VAM de svoje štampano kolo zaštitite od naponskih kratkotrajnih ekokove (tranzijenata) komponentime tipe TranZorb TM (Tranzient Absorber) koje pod oznakom P6KE proizvodi firme General Semiconductor (e I drugi) Ovi elementi, ne veći od obične diode mogu u kratkotrajnom vremenu de apsorbuju tranzijente snage reda 600W, e proizvode se za različite napone i različite energije

Priprema: Blažimir P. Miše, dipl ing.

USIJANI JOYSTCIK

Dobili smo nekoliko pisama u kojima se žalite da su igra u našem časopisu zapostavljene, da koncepcija rubrika nije baš najbolja, itd Odgovaramo svima o jednom trošku

Koncepcija igara, istina, još uvek nije ustaljena, pa kvalitet rubrika varira od broja do broja, all polako počinja da poprima konačan oblik. Igre koje predstavemo u časopisu su, obično, još nepoznate kod nas, i mi ne želimo da detaljno ulazimo u opise potpuno novih igara šta će ostati igračima ako „Im mi unapred sve otkrijemo '

Demon and Soft nas pita šta se radi u igri „Sex games" Uzgred on pita još stvari all je ovo najbolje pitanje E, pa lepo U toj igri roda nosi bebe preko kupusišta a ave u sv/hu obrazovanje malih demona Ahmm pitaj nekog od tvojih drugara, ako na znaš sam

Vesić Aleksandar iz Bora nam šalje svoja iskustva iz „Elita –

Vojislav Ristić iz Zemuna pita zašto zapostavljamo Igru „Tau Ceti", i čemu služi notes za beleške, kao i kako da spaja delove hladnjaka

Dragi Vojkana i mi amo se već zabrinuli zašto sa niko na javlja u vezi ova brilijantna igra. Grafika u njoj je toliko bolja od ona u „Elitl", koliko je „amiga bolja od „apektruma" O K

Notes ti služi za beleženja gredova kroz koje si prošao Seti se kako si manipulisao delovima pucii u „Impossible mission , pa to primani i za spajanja delova hladnjaka (/oda) Važno je da snimaš status u svakom mstu, jer, ako te unište /oboti, kreće iz početka U tome ti slabo pomaže POKE, koji je skoro objavljen u jednoj našoj reviji, jer blokira LOAD opciju I tako postaja beskoristan

Milan Stojković iz Zrenjanina plše da se najviše igra sa „Turbo Espritom I da ga interesuje nekoliko stvari

1 Kakav je to zvuk koji se čuja kada nalelu na pešaka?

2 Zašto se igra završava kada pobija sve dilera I zarobi oklopna kola?

3 Kako da iskoristi onih 30 milja (kolomatara?) na brojčaniku brzinomera? Maksimalna brzina je 150, a plše 180.

Evo i odgovora

1 Žalosno, all trebe da znaš da se pešaci, NE gaze A to što čuješ to oni vrište od bola I straha –

2 Igra se završava jer novi praprodavci nemaju zbog čega da dođu u taj grad

3 Nikako, to je štoa koji se i Inače koristi na pravim automobilima Na primer koji običan „kac može da ide 160 km/h?!

Smetramo da ste ovoga puta bili vrlo lenji na pisenja što, se lepo može videti iz rubrika Zato, olovke u šake I pište nam o svojim iskustvima Ne očekujte odgovor na baš svako pitanja jer se ni naši seradnici na igraju baš stalno I se svim igrama

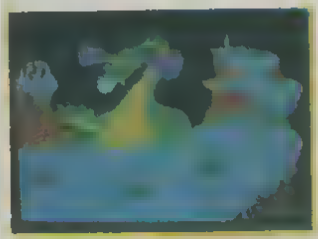
Čip i „Sedmarica"

u saradnji sa radio-emiterom „Čip i sedam jarića" koja se emituje svake subote na talasima Radio-Beograda u 12 časova

Komodor

DRAGON'S LAIR
(ZMAJEV BRLOG)

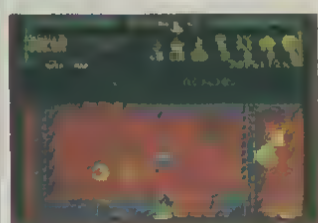
Software Projects



Dok Svati Đorđe ubiva aždahu u Ajsiju 212 i dok vaši dokoni prijatelji ganjaju ale vama se zalomio zmaj Dedeli žutooki zmaj koji je oteo vašu dragu i odveo je u svoj brlog iza sedam gora sedam mora i sedam nivoa Vi naravno pripasujete mač i krećete u pustolovinu Put vas vodi do podzemnih tamnica kuda se spuštate srednjovekovnom varjantom tifla predmatom veoma sličnim poklopcu za bačvu Dalje idete Hodnikom lobanja pa kroz suroviju verziju Kolosseuma gde se gerzanski prebacujete konopcima iznad razbukljale vatre pa preko krovova nekih ultramodernih građevina od plave cigle i tako stvari teku po rečima Meka Dizdara- sva lude i sva hude dok na slignale (ako uopšte slignete) do zmaja i ustanovite da ga i nije leško ubiti U poređenju sa raznovrsnim aspidama koja su vas dotle omatala on je maltene jagnjašce Prošujata se pored njega prikoljete ga mačem koji vas čeka na kraju puta — i ako mislite da je tu priči kraj u pravu ste.

DRUID

Firebird

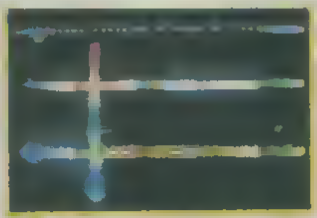


Druidi će se osloboditi karakondžula koja su im okupirale zemlju ako se probijate kroz osam nivoa i uništite čelirl lobanja Niste golorukl imate na raspolaganju sedam vrste čini Morate ih koristili strateški inače od celebadanja (i vašeg života) nema ništa Pomoću njih dobijate vatru, vodu struju Golema haos, neviđljivost i ključ Vatra uništeva skoro sva neprijatelje struja takođe ali manje efikasno voda deluje samo na odabrana Golem vam je telsana garda i uništeva ava đudmana haos uništeva lobanje (i ava ostelo što se zatekio na ekranu) pa mu ni to nije malo nego vam još obnavlja anargiju. Nevidljivost i ključ jsani su na prvi pogled imate samo jedan život kao i u stvarnosti od koje ste pobegli u ovu igru

Ako vam nije jasno otkud elektricitet u davnim mitskim vremenima ili kako je Golem stigao iz jevrejskih mitova a keltske pripremite se za razočaranje Odgovor na zna niko Ne budite kartezijanac

HYPERFORCE
(HIPERSILA)

Arlelasoft



Ako ste videli makar jednu Kleovu sliku setićete je se odmah Ovakve bravure se nijenesma umeo je (dosad) samo on

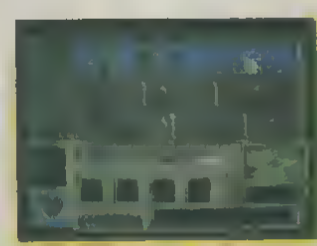
Ako ste makar jednom bili izpaljeni iz puška setićete se i tog doživljaja Brzina kojom ste tada leteli nije ni prinail brzini Hipersile

Sva ostalo biće vam novo Naći ćete se među apstrakinim oblicima upravljaćete oštećenim atartronom (zvezdotronom ake vam gravod ište objašnjava) skupljaćete mate kontejnere pucati u zidove-smetala ili ih vući unaireg tražićete predmete koji vam produžavaju život izbegavaćete predmate koji ga skraćuju — i lude se provoditi ako volite brzinu Drugog lonako nema ničeg ni druida, ni druida, ni duhova ni robota, ni čudovišta, ni karatista ni vitezova ni kostura ni laserskog oružja ni raketa

Odmoriće se

TIME TRAX
(VREMENSKI TOKOVI)

Mind Games



Komplikovano do zla boga

Pulujete kroz vrema de pokuplte razbacane dragocenosti i vratite ih na mesto iražite čelirl pločice ispisane runama (tek kad ih nađete spasli ste svat) baratate svakojakim oružjem boreći se protiv svakojake gamadi ukratko potucale se od nemila do nedraga kroz dvadesel jednu vremensku zonu (tri ekrana pute sedam zona) Uz to svaki ekran ima po tri nivoa

Tako vam i ireba kad hoćete da spasavate čovečanstvo

Gotovo polovinu ekrana zauzimaju indikatori ikonostas digitalni sat ležište za one četiri pločice sa runama itd Moraćete redovno ulaziti u spisak opcija da biste se snašli u igri i dobro pamiti da li ste ušli u Juče ili u Sutra ili nešto gde obojega Može vam se desiti da i u stvarnom životu primetite da je svanulo Sutra dok ste se probijali kroz igru započetu Sinoć

KNIGHT GAMES
(VITEŠKE IGRE)
English Software

Ovde bi briljirao Dajli Tompson da je kojim slučajem živeo u srednjem veku i da je bio vlastelin Doduše nije reč o dekatlonu nego o oktoatlonu ili osmoboju mačevanje 1 luče moćama gađanje sirelom (u drvene konje ne u protivnika) mekijanje buzdovanima cepanje helebardama mačevanje 2 gađenja samostrelom i sečenje bojnim sekirama Od vas se traži da budete što sličniji Milošu Vojinoviću

Ambijenti se menjaju da vam ne bude dosadno umesto sata služi zapaljena sveća (vaš se život gasi zajedno s njom) a neki položeni rambovi — ili su to dijamanti? — obaveštavaju vas o energiji vašoj i protivnikovoj

Na marljivo naslikanoj pozadini vitezovi su animirani vešto a nacrtani načinom srednjovekovnim to jeste nezgrapno Živopisac — videćete to dmah — nije mi prinati svojim studeničkim kolegama

Spektrum

EQUINOX
(RAVNDDNEVICA)
Mikro-Gen

Splet hodnika pun je kanistera U kanisterima je nešto nuklearno Kad istekne vrlme to Nešto

ekspiodira Vaše je da blagovremeno posklanjate kanistere Ometaju vas razne karakondžule

To je sve

Zašto onda (RAVNODNEVICA? Tko zna ah nitko niiko ne zna) krhko je znanje odgovara Dobrila Cesarić Ako ste se već našli u hodniku ne plašite se niša nego tražite propusnice i bušilice dižite smeće u vazduh idite iz jednog nivoa u drugi (i nairsg) pucajte iz sve snage istražujte koji su predmati korisni na kom nivou plaćajte plavim žetonima za vožnju koristite teleporte ustupite mesto starijim osobama i mahite na iznenadna kočenja

Na izvestan posredan način uverićete se da je daponovanje nukiearnih otpadaka velika glevobolja i da se irabe prikiadni čistoj i besplatnoj solarnoj energiji

DYNAMITE DAN II
(DAČA DINAMIT II)
Mirrorsoft

Igrati kompjuterska igra znači radiii Sizifov posao Taman ste u prvom DD likvidirali ludog naučnika — kad na lezi vraže pojavi se DD II i vi sa užasom videsia da je on vaskrsnuo Sad ga morate ganjati po osem osirva s svako se osirvo prosiire na 24 ekrane a na svekom ekranu vreba vas insta smrti

ali se ipak verujem nečete plaiii Ko me gurnu? Scenario je opak, ali maštovi morate bili veoma domišljati Uvrnute i siiski čisio izvedena scenografije pokazuje da imame posla sa ajajnim crtačem i koloristom

Na sedam osirva treba doneti ploču do džuboksa i odsvirati je potom naći cepelin napuniti ga porivom i odmagiiii na osmom ćete džu-boks Ići u vazduh i vratiti se u cepelin za iri minuta inače ginete Proiiv naučnika prepreka i kara kondžula pomažu pre svega lamne naočari baterijska lampa (da vam aspide na bi krale predmate) i mikser za hranu Igra je pune raznih predmeta neki su vam korisni a neki ne baš kao u pravom životu

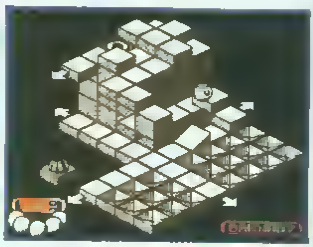
RETURN TO ITHACA
(POVRATAK NA ITAKU)
Atlantis

Oni koji su pročitali Odiseju imaju prednost napredovaće kroz ovu avanjuru na prijajući se ko su Lastrigonci ili šia je Kirkin mali hobi S druge strana svaka adaptecija po logici stvari podrazumeva izvesno odsiupanje od originala — a ovde ima izmena posle kojih ni sam Homer ne bi mogao prepoznati svoj spev Prema tome budite pažljivi ne čudite se što ćete kod Lotofaga morati ući u lavirint i odanda uzeti bure vina i na pokušajte da Polifemu dodate glave Odiseja vim načinom

Ima i detalja koji se u Odiseji — i u stvarnom životu — podrazumevaju ali ovde ne Tako kad negde pristanete ne zaboravite da otkucate DROP ANCHOR ij SPUSTI SIDRO — inače će vam brod otploviti nekud bestrage, pa ćete igru izgubiti

Ako Posejdon nadjača Atenu pa ne stignete na Itaku ne tugujte Penelopa je lonako dvadeset godina starija no što je bila kad ste krenuli u trojanski rat.

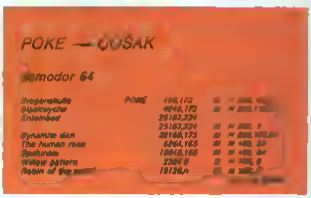
BOBBY BEARING
(KIĆA KUGLICA)
The Edge



O da Još jedna varijante na temu pustolo- klikar Otkako je SPINDIZZY povukao nagu (ako se to za kuglu može reći), nakotilo se Kreta Baunderš koliko vam duša hoće

Ovog puta jedna od onih kuglica kakve se prava u IKL-u poiazi na put da nađe četiri svoja brata i jednog rodaka sva petorica naime čame u ropstvu zlih kugličnih ležajeva Radnja se dešava u apsiraktnom trodimenzionalnom svetu kocki paraleiopipeda i prizmi Nade se i pokoji magnel kad on povuče kuglicu na nadajte se dobru Treba pomarati kockice čuvati se cigala koje padaju s neba propadati kroz šahtova na itd sve je to već viđeno

Da li? Ovakvu grafiku nismo videli Sjajna je Kad zatreba daje nam ulisak savršene sferičnosti — što pri rezoluciji kakvu ima spektrum nije nimalo lak posao Animaciji takođe svaka čast Težini još više da se ova igra odigra do kraja treba zaista imati kliker



Vladimir Stojiljković

POZIV NA PRETPLATU

Već 15 godina vaš pouzdani informator o nauci i tehnici kod nas i u svetu

POPUST 20%

PRETPLATA JE NAJBOLJI, NAJSIGURNIJI I NAJJEFTINIJI NAČIN NABAVKE NAŠEG ČASOPISA

ISKORISTITE SPECIJALNI POPUST ZA GODIŠNJU PRETPLATU I ISTOVREMENO SE ZAŠTITITE OD DALJIH POSKUPLJENJA

PREDNOSTI PRETPLATE
- manje cene (2.000 umesto 2.400)
- garantovane cene
- sigurna nabavka
- dostave na kuću

KADA SE PRETPLATITE, NEĆETE VIŠE MISLITI NA „GALAKSIJU"; ONA ĆE MISLITI NA VAS!
„Galaksije" je veše najveće, najpouzdanije i najlepše enciklopedije nauke i tehnike. Za petnaest godina izlaženja na prosečnom tiražu od 55.000 štampano je ukupno blizu deset miliona primeraka časopisa. Objavljeno je približno 12.000 članaka i otprilike 25.000 crno-belih i kolor ilustracija — od čega bi mogio da se načini oko 60 ilustrovanih monografija o nauci i tehnici.

NAUKA I TEHNIKA OD KAMENOG DOBA DO KOSMIČKE ERE

VAŠ VODIČ KROZ SVET NAUKE